Fachhochschulstudiengänge Burgenland GmbH
Steinamangerstraße 21
A-7423 Pinkafeld

Das Produktdatenmodell der Industry Foundation Classes (IFC) in der thermischen Gebäudesimulation am Beispiel RIUSKA

Diplomarbeit
zur Erlangung des akademischen Grades
Diplom-Ingenieurin (FH)
für technisch-wissenschaftliche Berufe

Betreuer: Prof. (FH) DI Dr. Wilhelm Zapfel
Eingereicht von: Carina Sagerschnig
Personenkennzeichen: 0210002031
Datum: Juni 2006

Vorwort

When we mean to build,
We first survey the plot, then draw the model;
And when we see the figure of the house,
Then must we rate the cost of the erection;
Which if we find outweighs ability,
What do we then but draw anew the model.

William Shakespeare
King Henry IV
circa 1597

Die Anregung zu dieser Diplomarbeit stammt von meinem Betreuer DI Dr. Wilhelm Zapfel. Ihm sei auf diesem Wege für die Unterstützung bei der Erstellung dieser Arbeit gedankt. Mein Dank gilt auch Herrn Werner Scheinbogen (Data Design System) und Herrn Antti Karola (Olof Granlund Oy) für die geduldige und ausführliche Beantwortung meiner vielen Fragen. Gedankt sei auch den Herrn Dr. Günther Wehrberger und Franz Stoiber des EDV – Labors an der TU Wien für eine anregende Diskussion über die IFC – Schnittstelle.

Abschließend möchte ich mich für die Geduld und Unterstützung bei meiner Familie, meinen Freunden und meinen Kollegen im „Diplomandenbüro" bedanken.

Carina Sagerschnig

Kurzfassung

Die rechnergestützte Projektabwicklung ist aus der Bauindustrie nicht mehr wegzudenken. Der Qualität des Datenaustausches kommt daher immer größere Bedeutung zu. Neben den verbreiteten CAD – Austauschformaten DXF und DWG wird eine neue Generation des Datenaustausches entwickelt. Produktdatenmodelle ermöglichen die Übertragung von integrierten Gebäudemodellen. In diesem Zusammenhang wird von Softwareinteroperabilität gesprochen. Ein Produktdaten-modell enthält neben geometrischen und topologischen Daten auch semantische Informationen. Dazu zählen u. a. Materialien, Materialbeschreibungen, Raumdefinitionen, Nutzungsrechte oder Zeitpläne.
Für die Erstellung dieser „intelligenten" Gebäudemodelle wird objektorientierte Software verwendet. Grundlagen der objektorientierten Programmierung, die durch Klassenbildung und Vererbungsstrukturen charakterisiert wird, sind Produktdaten-modelle. Die ISO 10303 (STEP) definiert einen allgemeinen Standard für den Datenaustausch mit Produktdatenmodellen.
Seit 1995 entwickelt die International Alliance for Interoperability (IAI) das Produktdatenmodell der Industry Foundation Classes (IFC). Dieses Modell ist herstellerunabhängig und richtet sich an die Bauwirtschaft. Einsatzbereiche der IFC sind in allen Bereichen der Bauwirtschaft, besonders in der Planungsphase, zu finden. Das IFC – Modell ist sehr umfangreich und deckt mit einer komplexen Datenstruktur verschiedene spezielle Anwendungsbereiche ab. Der Aufbau der IFC orientiert sich teilweise an den Standards der ISO 10303. Internationalität, Herstellerunabhängigkeit und die laufende Weiterentwicklung der IFC – Schnittstelle führt zu einer fortschreitenden Etablierung des Modells. Für den Datenaustausch stehen sowohl Dateiformate als auch Modell Server zur Verfügung.

Jede thermisch – energetische Gebäudesimulation basiert auf einem Modell. Die Genauigkeit des Modells, d. h. die Nachbildung der Wirklichkeit, wirkt sich entscheidend auf die Qualität der Simulationsergebnisse aus. Die Erfassung der Eingabedaten ist daher von besonderer Bedeutung. Traditionell werden diese Daten aus einer Vielzahl an Quellen zusammengetragen und händisch in ein unabhängiges Simulationsprogramm eingegeben. Der Zeit- und Ressourcenaufwand für die Modellerstellung überwiegt den Aufwand der eigentlichen Simulation. Vereinfachungen in der Erfassung des Gebäudemodells wirken sich positiv auf den gesamten Simulationsprozess aus. Das Hauptaugenmerk bei der Implementierung der IFC – Schnittstelle in Gebäudesimulationsprogramme liegt daher in der Über-tragung der Gebäudegeometrie.
Das Simulationsprogramm RIUSKA von Olof Granlund Oy wird für die dynamische Simulation von Behaglichkeit und Energieverbräuchen in Gebäuden verwendet. Das Programm ist eine benutzerfreundliche, graphische Oberfläche für den Simulationskern DOE-2. Über eine IFC – Schnittstelle werden geometrische und nicht – geometrische Daten des Gebäudemodells übernommen. Die aufwändige Nachbildung des Geometriemodells entfällt. Ergebnisse der Simulation können in das IFC – Gebäudemodell übertragen und von anderen Anwendungen weiterverwendet werden. Ein Praxistest mit fünf IFC – Beispielmodellen dokumentiert die derzeitigen Möglichkeiten der Übertragung von IFC – Modellen in RIUSKA.

Stichworte: Produktdatenmodelle, Gebäudesimulation, IFC, RIUSKA, DOE-2

Abstract

Computer aided project management is essential to the building industry. The quality of data exchange becomes increasingly important. In addition to widely used exchange formats like DXF and DWG a new generation of CAD data exchange is developed. Product data models enable the transmission of comprehensive, virtual building models. This is also called software interoperability. A product data model contains geometric and topological data as well as semantic information. This includes material description, room types, copyrights and schedules.
Object – oriented software is used to create "intelligent" building information models. Object – oriented programming, which uses characteristic elements like grouping and inheritance, is based on product data models. ISO 10303 (STEP) defines general standards for using product data models.
The International Alliance for Interoperability (IAI) started in 1995 the development of the Industry Foundation Classes (IFC) product data model. IFC are a non – proprietary product data model used especially in the building industry. There are plenty of application areas for IFC, mainly during schematic design but also for facilities management. The IFC data structure is extensive and covers different, detailed domains of the building industry. Partially, the STEP standards are used. Since IFCs are continuously developing, non – proprietary and international, the model is widely established. Both physical file exchange and IFC model servers are used for data exchange.

Building energy performance simulations are always based on building models. The accuracy of the model, meaning the correctness of recreating reality, is essential for high quality results. Therefore, acquisition of input data becomes particularly important. Traditionally, input data is collected from a variety of sources and manually input in a standalone simulation tool. Generating the input model often exceeds time and resources needed for simulation runs and analysis. Simplified access to existing building models positively affects the whole simulation process. In the field of building energy performance simulations IFCs are mainly used for importing building geometry into simulation tools.
RIUSKA is an energy performance simulation software tool by Olof Granlund Oy. It is used for comfort and energy simulation. RIUSKA is a graphic user – interface to the simulation core of DOE-2. To import geometric and semantic data of a building model an IFC interface is used. Simulation results can be written in the building model and thus are available for reuse by different software applications. To demonstrate current IFC capabilities five examples of IFC building models are imported to RIUSKA.

Keywords: Product Data Models, Building Energy Performance Simulation, IFC, RIUSKA, DOE-2

Inhaltsverzeichnis

1 EINLEITUNG 1

1.1 AUFGABENSTELLUNG 1
1.2 ZIELE DER DIPLOMARBEIT 1
1.3 VORGEHENSWEISE 1

2 DATENAUSTAUSCH IM BAUWESEN 2

2.1 PROBLEMATIK 2
2.2 VORHANDENE DATENAUSTAUSCHFORMATE 2
2.2.1 DXF und DWG 3
2.2.2 STEP-CDS 3
2.3 INTERNATIONALE NORMEN UND RICHTLINIEN 4
2.3.1 ISO 10303: STEP 4
2.3.2 VDI 6027: Anforderungen an den Datenaustausch von CAD Systemen 4
2.3.3 VDI 3805: Produktdatenaustausch in der TGA 6
2.4 ÖSTERREICHISCHE RICHTLINIEN 7

3 GRUNDLAGEN DER PRODUKTDATENMODELLIERUNG 8

3.1.1 Objektorientierte Softwareentwicklung 8
3.1.2 Produktmodellierung 9
3.1.2.1 Form der Produktmodellierung 12
3.1.2.2 Modellarchitektur 12
3.1.2.3 Unterschiede zu Geometriemodellen 13
3.1.3 ISO 10303: STEP 14

4 DAS PRODUKTDATENMODELL DER IFC 18

4.1 ZIEL UND FUNKTION 18
4.2 EINSATZBEREICHE 18
4.3 DATENSTRUKTUR DER IFC 19
4.3.1 Übersicht des Datenmodells 20
4.3.2 Ressourcen 21
4.3.3 Generische Datenstruktur 21
4.3.4 Konkrete Datenstruktur 23
4.3.5 Aufbau einer IFC – Datei 24
4.3.6 ifcXML 26
4.3.7 Bewertung des IFC – Modells 28
4.4 IMPLEMENTIERUNG 28
4.4.1 Verschiedene Arten von IFC – Kompatibilität 29
4.4.2 Methoden des Datenaustausches 30
4.4.3 IFC – Modell Server 30
4.4.4 Middleware Tool: BS Pro COM – Server 32

4.4.5 Implementation Views 33
4.5 BISHERIGER ENTWICKLUNGSVERLAUF 34
4.6 WEITERENTWICKLUNG DES MODELLS 35
4.6.1 Extension Projects 35
4.6.2 Zugriff auf IFC – Modell Server mit SABLE 35
4.6.3 Building Information Modeling – BIM 36
4.6.4 IFC und VDI 37
4.7 ANWENDUNG 38
4.7.1 IFC – kompatible Applikationen 38
4.7.2 Anwenderhandbuch der IAI 38
4.7.3 Analysewerkzeuge 38
4.7.4 Praxiserfahrungen 39
4.7.4.1 Pilotprojekt: PM4D 39
4.7.4.2 Pilotprojekt: Multi – disziplinäres Design – Studio 41

5 IFC IN DER THERMISCHEN GEBÄUDESIMULATION 43

5.1 EINLEITUNG 43
5.2 RICHTLINIEN DES VDI 43
5.2.1 VDI 6020: Anforderungen an Rechenverfahren zur Gebäude- und Anlagensimulation .. 43
5.2.2 VDI 6021: Datenaustausch für die thermische Lastberechnung von Gebäuden 44
5.3 DAS GEOMETRISCHE GEBÄUDEMODELL DER SIMULATION 47
5.3.1 Bedeutung des Geometriemodells 47
5.3.2 Nachbildung der Gebäudegeometrie 49
5.3.3 Import bestehender Gebäudegeometrien 49
5.3.3.1 Direkte Schnittstelle 49
5.3.3.2 Graphische Benutzeroberfläche 50
5.3.3.3 Softwareinteroperabilität 51
5.4 BEDEUTUNG DER IFC FÜR DIE THERMISCHE SIMULATION 51
5.4.1 Einsatzbereiche 51
5.4.2 Das Gebäude aus Sicht der thermischen Simulation 52
5.4.3 IFC – Kompatibilität 53
5.4.4 IFC – kompatible Simulationsprogramme 55

6 GEBÄUDESIMULATION MIT RIUSKA 56

6.1 EINLEITUNG 56
6.2 ANWENDUNGSBEREICHE VON RIUSKA 57
6.3 SIMULATIONSKERN DOE-2 57
6.3.1 Programmüberblick 58
6.3.2 Aufbau 58
6.3.3 Wetterdaten 60
6.3.4 Berechnungsmethoden 61
6.3.5 Anwendungsbereiche 62
6.3.6 Weiterentwicklung 62
6.4 DOE-2 UND RIUSKA 65

6.4.1 DOE-2 in RIUSKA 65
6.4.2 DOE-2 ohne RIUSKA 66
6.5 IFC – SCHNITTSTELLE 67
6.6 THERMISCHE SIMULATIONEN IN RIUSKA 68

7 IFC – MODELLE IN DER PRAXIS 70

7.1 BEISPIELPROJEKTE DER IAI 70
7.2 ÜBERPRÜFUNG DER IFC – MODELLE 72
7.3 IMPORT DER IFC – MODELLE IN DDS 74
7.4 ERSTELLUNG EINES NEUEN IFC – MODELLS IN DDS 78
7.5 EINSATZ DER IFC – MODELLE IN DER THERMISCHEN SIMULATION MIT RIUSKA 79
7.5.1 Allgemeiner Umfang des IFC – Imports in RIUSKA 79
7.5.2 Import der IFC – Modelle 80
7.5.3 Informationsübertragung in das IFC – Modell 86
7.6 SCHLUSSFOLGERUNG 89

8 ZUSAMMENFASSUNG 91

9 GLOSSAR 97

10 LITERATURVERZEICHNIS 107

11 ABBILDUNGSVERZEICHNIS 112

12 TABELLENVERZEICHNIS 114

1 Einleitung

1.1 Aufgabenstellung

Die Aufbereitung der Eingabedaten für eine thermische Gebäudesimulation stellt einen hohen Zeit- und Kostenfaktor dar. Gebäudepläne und thermische Angaben zu Bauteilen müssen von Hand in das Simulationsprogramm übertragen werden. Objektorientierte Anwendungen und Softwareinteroperabilität ermöglichen die verlustfreie Übertragung von Gebäudemodellen zwischen Architekt und TGA – Planer über die Geometrie hinaus.

In dieser Arbeit soll das Produktdatenmodell der *Industry Foundation Classes* (IFC) auf Aufbau und Funktionsweise untersucht werden. Derzeitige Einsatzmöglichkeiten, Leistungsgrenzen und zukünftiges Potenzial der IFC sollen gezeigt werden. Die Anwendung des Modells wird im Speziellen für die thermische Gebäudesimulation betrachtet. Hierzu wird das IFC – kompatible Simulationsprogramm *RIUSKA* von Olof Granlund Oy verwendet.

1.2 Ziele der Diplomarbeit

- Überblick der aktuellen Möglichkeiten des Datenaustausches im Bauwesen,
- Erläuterung von Zielen, Aufbau und Funktionen der IFC,
- Beschreibung von Einsatzmöglichkeiten der IFC,
- Betrachtung des Gebäudemodells und seiner Bedeutung für die thermische Gebäudesimulation,
- Möglichkeiten und Auswirkungen der IFC auf die thermische Gebäude-simulation,
- Beschreibung der Verwendung einer IFC – Schnittstelle an Hand des Simulationsprogramms RIUSKA.

Im Rahmen dieser Diplomarbeit wurde ein Handbuch für RIUSKA erarbeitet, das unabhängig von der Diplomarbeit zu verwenden ist. Das Handbuch enthält eine Übersicht der Programmfunktionen, eine Beschreibung der Simulationsmöglichkeiten sowie einen Theorieteil, der in das Thema der IFC und das Simulationsprogramm DOE-2 einführt.

1.3 Vorgehensweise

Für die Einarbeitung in das Produktdatenmodell der IFC wird eine Literaturrecherche durchgeführt. Über die Literatur hinaus werden die IFC – kompatiblen Programme Data Design System (DDS) und RIUSKA für einfache thermische Testsimulationen und die Erstellung von IFC – Testmodellen verwendet.

2 Datenaustausch im Bauwesen

Dieses Kapitel behandelt den aktuellen Stand des CAD – Datenaustausches in der Bauindustrie. Wichtige Austauschformate sowie nationale und internationale Normen und Richtlinien werden besprochen. Die VDI – Richtlinien 6027 und 3805 stellen Anforderungen an den CAD- und Produktdatenaustausch. Ein internationaler Standard für den Datenaustausch mittels Produktdatenmodellen wird in der ISO 10303 definiert. Auf nationaler Ebene werden Richtlinien von Bund und Ländern betrachtet.

2.1 Problematik

Die Bauinformatik ist eine wichtige Grundlage des Bauingenieurwesens. Sie beschäftigt sich mit dem Einsatz des Computers als Rechenmaschine, Datenträger, Darstellungsmedium sowie mit Computerarbeitsplätzen und Netzwerken (Beuke et al., 2000). Für die Baupraxis ist die rechnergestützte Bearbeitung eines Bauvorhabens unerlässlich.

Von verschiedenen Herstellern wurde eine Vielzahl an Programmen entwickelt. Manche Anwendungen bearbeiten mehrere Schritte im Projektverlauf, andere sind auf einen Arbeitsschritt spezialisiert. Im Laufe des Projektfortschritts steigen die Datenmenge und die verwendete Anzahl an Computerprogrammen. Inkompatible Softwareanwendungen und herstellerspezifische Datenformate führen dazu, dass einmal erstellte Datensätze nicht weiterverwendet werden können. Im Laufe der Planungsphase wird die Geometrie des Gebäudes sieben bis achtmal nachgebildet und in verschiedensten Anwendungen neu gezeichnet (Bazjanac, 2001). Bereiche wie Statik, Elektroplanung, Installation, Lebenszyklusberechnungen, Lichttechnik und Kostenschätzung benötigen die Gebäudegeometrie als Grundlage. Eine manuelle Neueingabe der Daten bedarf an Zeit und Personal und ist fehleranfällig. Besonders der Datenaustausch von Gebäudemodellen, d. h. der Austausch von geometrischen und nicht – geometrischen Daten des gesamten Gebäudes, ist für die Bauwirtschaft zunehmend von Bedeutung.

2.2 Vorhandene Datenaustauschformate

Der Datenaustausch zwischen Anwendungen findet über Schnittstellen statt. Es wird zwischen individuellen und standardisierten Formen unterschieden (Dayal, 2004). Individuelle Schnittstellen sind genau auf die beteiligten Programme abgestimmt. Sie erlauben eine hohe Übertragungsqualität. Um den Datenaustausch zu realisieren, ist für jeden Portierungsweg ein Import- sowie Exportmodul zu entwickeln. Soll der Datenaustausch zu mehreren Programmen ermöglicht werden, werden für jedes Austauschszenario eigene Import- und Exportmodule benötigt.

Um den Entwicklungsaufwand zu verringern, aber gleichzeitig den Datenaustausch zwischen mehreren Anwendungen zu ermöglichen, wurden standardisierte

Schnittstellen eingeführt. Voraussetzung eines Standards ist die Einigung der Hersteller auf ein Austauschformat. Der Implementierung, d h. der Einbindung der Schnittstelle in die Software kommt große Bedeutung zu, da Ungenauigkeiten zu einem Datenverlust in der Übertragung führen.

2.2.1 DXF und DWG

Die meisten Daten werden immer noch in herstellerabhängigen Datenformaten gespeichert und übertragen. DXF (Data Exchange Format) ist ein Zeichnungsformat des Softwareherstellers Autodesk. Autodesk entwickelt u. a. AutoCAD und Architectural Desktop (ADT). Autodesk unterstützt DXF als offenes Austauschformat für 2D/3D – Zeichnungen. Es enthält Vektorgrafiken des Modells im ASCII – Format. Darüber hinaus können Strukturinformationen mittels Layern und Blockdefinitionen ausgetauscht werden.

DWG ist ein internes Zeichnungsformat von Autodesk. Da Autodesk der weltweit führende CAD – Entwickler ist, kommt dem Format besondere Bedeutung zu. Die Definition des DWG – Formats ist nicht frei zugänglich. Um DWG – Daten in anderen CAD – Anwendungen verarbeiten zu können, wurde 1998 die Open Design Alliance gegründet. Ziel der Allianz ist die Förderung von OpenDWG, einer offenen und herstellerunabhängigen Version des Autodeskformats (Open Design Alliance, 2006). Ihre Implementierung ermöglicht CAD – Anwendungen DWG – Daten zu lesen bzw. schreiben. Zu den Mitgliedern der Allianz gehören u. a. Graphisoft, Nemetschek und Bentley Systems. Autodesk ist kein Mitglied.

DXF und DWG beschreiben das geometrische Modell. Es können daher nur Zeichnungen und wenige Strukturinformationen übertragen werden. Informationen über das Produkt, d. h. das Gebäude, sind nicht enthalten. Bauteile, ihre Konfiguration oder Beziehungen zwischen Bauteilen, z. B. die Zuordnung von Wänden zu einem Raum, sind nicht vorhanden.

2.2.2 STEP-CDS

1997 startete in Deutschland eine Initiative zur Umsetzung des STEP – Standards der ISO 10303 (siehe Kapitel 2.3.1 und 3.1.3) mit dem Ziel das Niveau des Datenaustausches im Industriebau qualitativ zu verbessern (Haas, 1999). STEP – CDS wird von den Marktführern der CAD – Systeme in der Baubranche unterstützt.

Das Leistungsspektrum von STEP – CDS umfasst:

- Modelldaten von 2D – CAD Geometrien
- Modellstrukturen (z. B. Layer, Gruppen)
- Administrative Daten (z. B. Version, Freigabevermerke)
- Sachdaten (z. B. Material, Leistungskennwerte, Klassifikation)
- Verweise auf externe Dokumente
- Technische Zeichnungen mit Symbolen, Bemaßung, Toleranzen, Schraffur, Beschriftung und Zeichnungslayout.

STEP – CDS ist eine Initiative zur Optimierung des herstellerunabhängigen 2D – CAD – Datenaustausches. Die Schnittstelle ist sehr leistungsstark, wenn auch stark auf den Maschinenbau ausgerichtet (Dayal, 2004).

2.3 Internationale Normen und Richtlinien

Nationale und internationale Normen und Richtlinien wurden entwickelt, um den unabhängigen Produktdatenaustausch auf hohem Niveau zu erleichtern. Programminterne Speicherformate sind von den Richtlinien nicht betroffen und werden weiterhin verwendet.

2.3.1 ISO 10303: STEP

Die ISO 10303 definiert einen Standard für den Datenaustausch mit Produktmodellen. Die Norm wird auch als STEP – Standard (Standards for the Exchange of Product Model Data) bezeichnet und richtet sich nicht nur an die Bauwirtschaft. Branchenübergreifende Grundlagen werden ebenso festgelegt, wie spezifische Produktdatenmodelle – und damit Datenaustauschstandards – für bestimmte Anwendungsbereiche. Aufbau und Ziele der ISO 10303 werden im Rahmen der Produktdatenmodellierung genauer erläutert (siehe Kapitel 3.1.3).

2.3.2 VDI 6027: Anforderungen an den Datenaustausch von CAD Systemen

Die Anforderungen an den Datenaustausch von CAD – Systemen werden in der VDI – Richtlinie 6027 festgelegt. Blatt 1, 1999 im Weißdruck erschienen, betrifft Konventionen für Gebäude und Gebäudetechnik. Schwerpunkt des Blattes sind die zumindest einzuhaltenden Layerstrukturen, um eine Weiterverwendung ohne Nachbearbeitung gewährleisten zu können. In der Layerstruktur werden u. a. Maßstab, Zeichensätze, Layertypen und Referenzen definiert.

In Blatt 2, erschienen im November 2005, wird der CAD – Datenaustausch in der Anlagentechnik behandelt. Ein strukturierter Datenaustausch sowohl innerhalb der TGA als auch zwischen TGA und anderen Beteiligten soll gewährleistet werden. Dazu wird ein Datenaustauschformat im STEP – Standard beschrieben. Der Informationsaustausch geht über den Austausch von Geometrie hinaus. Ziel ist eine umfassende Beschreibung der Anlage im gesamten Lebenszyklus. Wo vorhanden werden daher bereits existierende Datensätze referenziert. Die Zuordnung zu einem Gebäudemodell wird über eine Verbindung zum Gebäudemodell aus der VDI 6021 (siehe Abschnitt 5.2.2) realisiert. Informationen über Produktdaten werden aus der Produktbeschreibung nach VDI 3805 übernommen.

Die Schnittstelle nach VDI 6027 unterteilt eine Anlage in Anlagenkomponenten, die über ein Anlagennetz verbunden sind. Die Beschreibung der Anlagenkomponenten

umfasst eine lokale Zuordnung, die Beschreibung von Produkteigenschaften, Auslegungsparametern und eventuell Betriebsparametern.

Die lokale Zuordnung erfolgt über die Zuordnung zu einem Raum. Die Zuordnung des Raums zu einem Gebäudemodell ist in der VDI 6021 dargestellt (Abbildung 2.1).

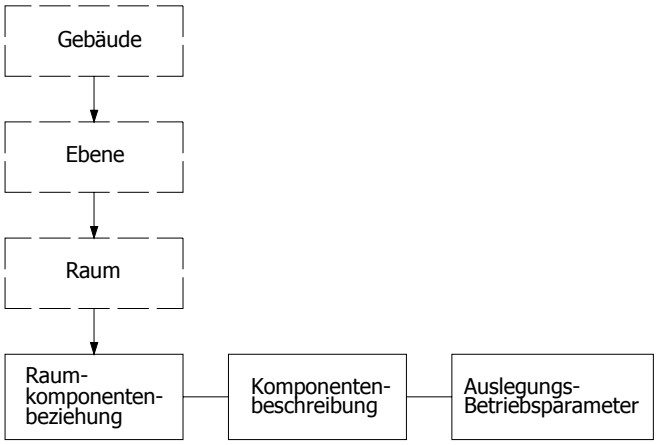


Abbildung 2.1: Struktur der lokalen Zuordnung von Anlagenkomponenten (nach VDI 6027)

Die Produktbeschreibung erfolgt durch eine TGA – Nummer, die bereits in der VDI 3805 definiert ist (siehe folgender Abschnitt). Durch die eindeutige Referenzierung können alle das Produkt betreffenden Daten aus der Produktdatenbank ausgelesen werden. Ein allgemeiner Datensatz ist für nicht in der VDI 3805 beschriebene Produkte enthalten.

Die Struktur des Anlagennetzes ist in Abbildung 2.2 dargestellt. Das Anlagennetz besteht aus einer Anlagengruppe, der eine Netzart zugeordnet werden kann. Der Anlagengruppe „Heizkreis" wird z. B. die Netzart „Vorlauf" zugewiesen. Das Netz besteht aus Teilstrecken, die sich durch einen konstanten Massenstrom im Auslegungsfall ergeben. Eine Teilstrecke wird durch Knoten begrenzt.

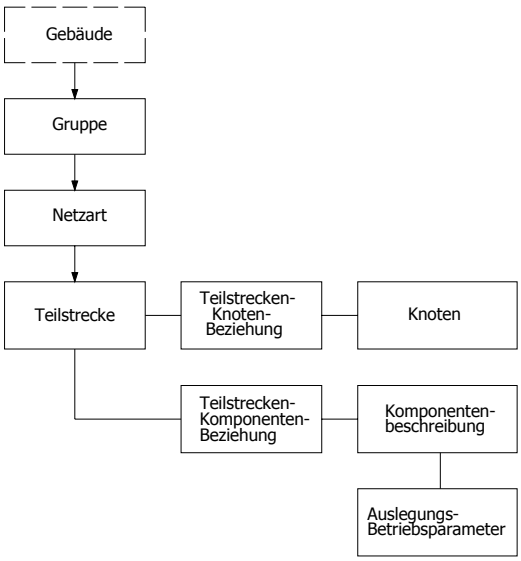


Abbildung 2.2: Allgemeine Struktur des Anlagennetzes (nach VDI 6027)

Eine weitere Unterteilung der Teilstrecken kann gegeben sein, wenn nach Eigenschaften unterschieden wird (z. B. Änderung der Wärmedämmung oder Umgebungstemperaturen). Es werden fiktive Knoten gebildet. In den Teilstrecken können Komponenten eingebaut sein.

Das Datenaustauschformat ist in Blatt 2 der VDI 6027 vollständig offen gelegt. Nach STEP ist ein EXPRESS – Schema und eine grafische Darstellung in EXPRESS – G für Anlagenkomponenten und Anlagennetze vorhanden. Weiters wird der Inhalt der Informationseinheiten beschrieben. Diese Richtlinie wird von den IFC abgedeckt (vgl. Abschnitt 4.6.4).

2.3.3 VDI 3805: Produktdatenaustausch in der TGA

Die VDI – Richtlinie 3805 stellt die Grundlagen des Produktdatenaustausches in der TGA dar. Der Weißdruck liegt seit November 2001 vor. Ziel der VDI 3805 ist eine einheitliche Darstellung der Daten gebäudetechnischer Produkte durch den Hersteller. Daten gleichartiger Produkte besitzen einen gleichen Aufbau. Hauptmerkmal des Datenaustausches ist die Kennzeichnung der Produkte durch eine sogenannte TGA – Nummer. Mit Hilfe der TGA – Nummer wird die Hierarchie der Produktstruktur abgebildet. Regelungen über den Austausch von Preisinformationen sind nicht Teil der VDI 3805.

Folgende Produktdaten werden durch den Hersteller zur Verfügung gestellt:

- Technische Daten zur Auslegung des Produkts und der zugehörigen Anlagen.
- Wertetabellen zur Erzeugung der vorgegebenen Geometrie. Beinhaltet sind eine 3D – Darstellung, eine 2D – Maßzeichnung sowie eine schematische Darstellung.
- Artikel-, Datanorm- und StLB – Nummern mit Textergänzung für die standardisierte Ausschreibung.

Die normierten Datensatzbeschreibungen ermöglichen CAD – Anwendungen den Einsatz von Produktdatenkatalogen verschiedener Anbieter ohne die herstellerabhängige Entwicklung einer Datenschnittstelle.

Ein Standardformat für den Zeichnungsaustausch ist im Rahmen dieser Richtlinie nicht gegeben. Geometriedaten und Stör- bzw. Bewegungskonturen werden in Form einer Wertetabelle gespeichert. Die Darstellung der Geometrie muss von CAD – Programmen übernommen werden.

Die Richtlinie ist in mehrere Blätter unterteilt. Blatt 1 enthält die Grundlagen des Produktdatenaustausches, das allgemeine Produktdatenmodell, die Datensatzstruktur des Modells und die Beschreibung der Geometriedaten. Da die Produktbeschreibung stark von der betreffenden Produktgruppe abhängt, werden in zahlreichen Folgeblättern spezielle Produktgruppen behandelt. Tabelle 2.1 zeigt eine Übersicht der bereits erschienen Blätter.

Tabelle 2.1: VDI 3805: Produktdatenaustausch in der TGA – Folgeblätter

Blatt	Produktgruppe	Ausgabedatum
Blatt 2	Heizungsarmaturen	Februar 2003
Blatt 3	Wärmeerzeuger	Juni 2004
Blatt 4	Pumpen	April 2005
Blatt 5	Luftdurchlässe	Jänner 1999
Blatt 5	Luftdurchlässe (Entwurf)	Dezember 2005
Blatt 6	Heizkörper	Mai 2004
Blatt 7	Ventilatoren	Juli 2005
Blatt 8	Brenner	Juni 2004
Blatt 9	Modullüftungsgeräte	April 2002
Blatt 10	Luftfilter	Juli 2003
Blatt 11	Wärmetauscher Fluid/Wasserdampf - Luft	Juli 2003
Blatt 14	RLT – Schalldämpfer (passiv) (Entwurf)	Mai 2005
Blatt 16	Brandschutzklappe	Juli 2003
Blatt 17	Armaturen für die Trinkwasserinstallation	November 2003
Blatt 19	Sonnenkollektoren	Februar 2006
Blatt 20	Speicher und Durchlauferhitzer	Februar 2006
Blatt 22	Wärmepumpen	Juni 2005
Blatt 23	Wohnungslüftungsgeräte	Mai 2005

Zurzeit werden u. a. die Produktgruppen Ausdehnungsgefäße, Wärmerückgewinnung und Flächenheizung/Kühlung bearbeitet. Geplant sind Folgeblätter u. a. für Split – Anlagen, Induktionsgeräte und Messeinrichtungen für die Heizkostenabrechnung (VDI Richtlinienausschuss 3805, 2006).

2.4 Österreichische Richtlinien

Verschiedene Richtlinien von Bund und Ländern beschäftigen sich mit dem CAD – Datenaustausch, z. B. die „Richtlinien CAD Hochbau" der Stadt Wien (Ausgabe 1998). Diese und ähnliche Richtlinien beziehen sich hauptsächlich auf die Austauschformate zweidimensionaler Pläne und die verwendete Layerstruktur.

2004 veröffentlichte das Bundesministerium für Landesverteidigung die „Richtlinie CAD Hochbau". Im Rahmen dieser Richtlinie werden verschiedene Grade der Lieferqualität unterschieden. Die höchste Lieferqualität entspricht einem digitalen Gebäudemodell im Format ADT 2004 der Firma AutoDesk. Im Vorwort der Richtlinie wird dieser Schritt als Vorbereitung auf den Einsatz der IFC als produktneutrales Datenaustauschformat gesehen.

3 Grundlagen der Produktdatenmodellierung

In den folgenden Kapiteln werden allgemeine Prinzipien eines Produktdatenmodells und der Aufbau des Produktdatenmodells der Industry Foundation Classes im Speziellen erläutert. Die Programmierung des IFC – Produktmodells ist für den Endnutzer IFC – kompatibler Software von geringer Bedeutung. In der Softwareentwicklung kommen Umfang und Aufbau der IFC hingegen eine große Bedeutung zu. Ein zu großer Programmieraufwand in der Erstellung einer IFC – Schnittstelle, die ein Programm IFC – kompatibel macht, wirkt sich durch Verzögerungen in der Produktentwicklung oder im Extremfall durch den Verzicht auf IFC – Kompatibilität indirekt auf den Endnutzer aus. Die Erläuterung des Aufbaus des Datenmodells erfolgt in dieser Arbeit aus zwei Gründen:

- Eine kurze, allgemein verständliche Beschreibung der Datenstruktur soll einen kleinen Einblick in das Gebiet der Bauinformatik geben. Für das tiefergehende Studium der IFC – Programmierung wird auf weiterführende Literatur verwiesen.
- Aus der Definition der IFC – Datenstruktur herrührende Möglichkeiten und Einschränkungen des Produktdatenmodells sollen erklärt werden. Entwicklungen des Modells und seiner Implementierung können leichter veranschaulicht werden.

3.1.1 Objektorientierte Softwareentwicklung

Die Grundlage von Softwareinteroperabilität in der Bauindustrie bilden objektorientierte Produktdatenmodelle. Objektorientierte Software basiert auf diesen Produktdatenmodellen. Abweichend von herkömmlicher Programmierung ist die sinnvolle Nachbildung von real existierenden Objekten der Schwerpunkt der objektorientierten Programmierung. Dabei werden physikalische (z. B. Türen, Wände) und abstrakte Objekte (z. B. Räume, Prozesse) nachgebildet. Diese Objekte besitzen Eigenschaften in Form von Attributen und Funktionalitäten.

Die Gliederung eines Problems in Objekte sowie die Merkmale und Beziehungen verschiedener Objekte sind Vorteile der objektorientierten Softwareentwicklung. Die Art der Programmierung bietet eine größere Übersichtlichkeit des Modells. Ein modularer Aufbau ermöglicht einfache Erweiterbarkeit, Wiederverwendbarkeit der Programme und Kompatibilität verschiedener Softwaremodule.

Die wichtigsten charakteristischen Elemente der objektorientierten Softwareentwicklung sind die Bildung von Klassen, Kapselung, Vererbung und Polymorphismus.

Die *Klassenbildung* ist das grundlegendste Prinzip der objektorientierten Softwareentwicklung. Auf ihr basieren Kapselung, Vererbung und Polymorphismus. Eine Klasse dient zur Beschreibung real existierender Objekte. Sie ist eine „Vorlage" und enthält Attribute zur Beschreibung von Objekteigenschaften eines bestimmten

Objekttyps. Klassen enthalten auch Methoden, die den Objekten zu Eigen sind. Methoden sind Funktionen und Prozeduren zur Beschreibung des Verhaltens von Objekten. Den Objekten der Klassen können in Anwendungen konkrete Werte zugewiesen werden, d. h. sie werden instanziiert. Die Form der Instanziierung einer Klasse ist in der Programmierung festzulegen.

Bei der *Kapselung* werden Attribute oder Methoden außerhalb einer Klasse unzugänglich gemacht. Sie werden eingekapselt. Drei Stufen der Kapselung werden unterschieden: Öffentlich und von außen zugänglich sind Attribute und Methoden der Deklaration *public*. *Protected* bedeutet, dass Attribute und Methoden in Unterklassen verfügbar sind. Bei der Deklaration *private* sind Attribute und Methoden nur in der Klasse in der sie definiert sind, zugänglich. Ein Vorteil der Kapselung liegt in der Kontrolle der äußeren Manipulation von Attributen. So können z. B. Eigenschaften, die einen bestimmten Wertebereich besitzen müssen, geschützt werden. Weiters sind gekapselte Objekte unabhängig von späteren Programmerweiterungen.

Real existierende Objekte können gleiche Eigenschaften besitzen, d. h. sie bilden eine Gattung. In der objektorientierten Softwareentwicklung werden Gattungen mit Hilfe der *Vererbung* nachgebildet. Oberklassen können ihre Eigenschaften (Attribute und Methoden) an ihre Unterklassen vererben. Man spricht von einer Spezialisierung in der Unterklasse. Eine Klassenhierarchie entsteht, wenn Unterklassen Vererbungen einer einzigen Oberklasse enthalten (einfache Vererbung). Eine netzwerkförmige Struktur entsteht bei der Mehrfachvererbung. Eine Unterklasse erbt in diesem Fall Eigenschaften mehrerer Oberklassen.

Ein weiteres Element der objektorientierten Softwareentwicklung ist der *Polymorphismus* (Vielgestaltigkeit). Trotz gleichen Namens können Funktionen je nach Situation verschiedene Aufgaben ausführen. Zu den wichtigsten Arten des Polymorphismus zählen Overloading und Overriding. *Overloading* bedeutet, dass Funktionen sich nicht durch den Namen sondern durch Art und Anzahl der Funktionsparameter unterscheiden. In Abhängigkeit des ausführenden Programms werden bestimmte Parameter übergeben und die Funktion ausgeführt. Abweichend davon wird beim *Overriding* erst während der Programmlaufzeit die auszuführende Funktion ausgewählt. Polymorphismus erleichtert dadurch die Erstellung des Gesamtprogramms (Haller, 1994).

Im Vergleich zur klassischen Programmierung bedeutet objektorientierte Softwareentwicklung einen Mehraufwand bei Neuentwicklungen. Durch Wiederverwendung und einfachere Wartung von Modulen entstehen allerdings Einsparungen. Auch die Einbindung und Anpassung verschiedener Softwareanwendungen untereinander wird vereinfacht.

3.1.2 Produktmodellierung

Produktmodelle ermöglichen die Kommunikation von Software gleicher oder verschiedener Anwendungsgebiete. In diesem Zusammenhang wird von *Interoperabilität* oder *Integration* gesprochen. Da im gesamten Lebenszyklus des Produkts kein Datenverlust auftreten soll, bewerkstelligt ein Produktmodell mehr als

reinen Datenaustausch. Es dient auch zur Archivierung und Speicherung von Produktinformationen.

Abbildung 3.1 stellt die Funktionen eines Produktdatenmodells dar. Für jeden Teilbereich müssen auf das Produktdatenmodell abgestimmte Methoden und Prozeduren entwickelt werden. So kann der Produktdatenaustausch die Verwendung eines bestimmten Austauschformats vorsehen. Archivierungsformate, Klassifizierungsmerkmale und Zugriffsschnittstellen müssen für die Produktdaten-archivierung festgelegt werden. Datenbanken und Datenverwaltungssysteme ermöglichen die Produktdatenspeicherung. Die Funktion der Produktdaten-transformation betrifft Transformationsmethoden für das Übertragen des Modells oder seiner Teilbereiche in andere Formate.

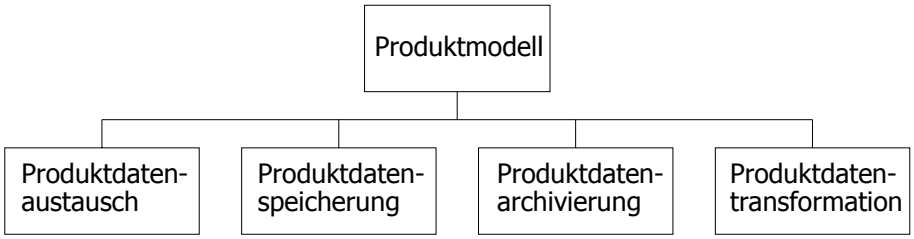


Abbildung 3.1: Funktionen des Produktmodells in der Datenverarbeitung (vereinfacht nach Grabowski et al., 1993)

Per Definitionem enthält das Produktmodell Daten des gesamten Planungsprozesses. Es basiert auf einer ganzheitlichen Betrachtung des Produkts, wobei zwischen den Teilmodellen Zusammenhänge hergestellt werden. Haller (1994) definiert das Produktmodell als „die Summe aller ein Produkt beschreibenden Daten".

Im Zusammenhang mit Produktdatenmodellen wird auch von der Übertragung semantischer Eigenschaften gesprochen. Semantische Eigenschaften bezeichnen Informationen unabhängig von Geometrie und Lage des Objektes, z. B. Material, Kosten, Listen, Zeitpläne oder Nutzungsrechte.

Das Produktmodell ermöglicht Interoperabilität durch verschiedene Kommunikations-arten. Die Datenübertragung kann über eine gemeinsame Datenbank oder einen Server erfolgen, auf den mehrere Anwendungsprogramme Zugriff haben. Unabhängig von Datenbanken und Servern kann die Datenübergabe auch durch den direkten Austausch einer Datei mittels ASCII – Schnittstelle erfolgen.

Hörenbaum (2002) beschreibt die Erstellung eines Produktdatenmodells nach folgenden Merkmalen:

- Der Anwendungsbereich des Modells muss genau abgesteckt werden. Durch Kenntnis der Abläufe in Planung und Fertigung kann das Modell auf die Anwender zugeschnitten werden.
- Die modellhafte Nachbildung der Prozesse erfolgt im Datenmodell, dem Kern jedes Produktmodells.
- Das Modell muss in Sichten, sogenannte Konformitätsklassen, unterteilt werden. Dadurch wird es Anwendungsprogrammen ermöglicht Teilbereiche

des Modells zu unterstützen. Konformitätsanforderungen definieren Regeln für die Unterstützung einzelner Sichten durch bestimmte Anwendungsprogramme.
- Das Produktmodell dient zur Beschreibung des Produkts. Um es für den aktiven Datenaustausch verwenden zu können, muss es an Anwendungssoftware angebunden werden, d.h. das Produktmodell wird implementiert.
- Durch eine abschließende Konformitätsprüfung wird festgestellt, ob Anwendungssoftware, die an das Produktmodell angebunden ist, den Regeln des Modells entspricht. Diese Prüfung garantiert dem Anwender die Funktion des Produktdatenaustausches und sollte von unabhängiger Seite durchgeführt werden.

Grundlage jedes Produktdatenmodells bildet die Spezifikationssprache, die es beschreibt. An die verwendeten Sprachmittel müssen nach Haller (1994) folgende Anforderungen gestellt werden:

- Eine eindeutige und verständliche Syntax muss verwendet werden, um falsche Interpretationen zu vermeiden.
- Da im Produktmodell das Produkt in einzelne Einheiten zerlegt wird, muss die verwendete Sprache die Definition solcher Grundeinheiten (Entities) ermöglichen.
- Zusammenhänge zwischen den Entities müssen erstellt werden können.
- Entities werden über ihren Namen beschrieben. Sie beinhalten beliebig viele Attribute verschiedener Typen.
- Zeichenketten, Fließkomma- und Integerzahlen sollten mindestens als Attributstypen angewandt werden können.
- Der Inhalt des Attributs muss durch einen Wertebereich definierbar sein.
- Ein Datentyp *Menge* sollte existieren, d. h. ein Objekt kann aus mehreren gleichen Objekten niedrigerer Stufe aufgebaut werden.

Die Anwendung von Produktdatenmodellen hängt letztlich von der Akzeptanz durch Software – Hersteller ab. Eine aufwändige Implementierung durch einen hohen Detaillierungsgrad des Modells schreckt ebenso ab, wie problematische Erweiterungen eines zu komplexen Modells.

Der Umfang einzelner Sichten (Teilbereiche) des Modells stellt eine weitere Problematik dar. Zu große Teilmodelle überschreiten die Anwendungsgrenzen vieler Anwendungsprogramme. Kleinere Sichten sind einfach zu implementieren, erschweren aber die Kommunikation verschiedener Programme. Ein Datenaustausch kann dann nur im begrenzten Bereich des Teilmodells stattfinden.

Modellarchitektur und Art der Modellierung spielen für die Akzeptanz eines Produktdatenmodells eine wichtige Rolle. Verschiedene Modellarten werden in den folgenden Abschnitten erläutert.

3.1.2.1 Form der Produktmodellierung

Jedes Modell orientiert sich an der Realität. Das Datenmodell nimmt das menschliche Denken zum Vorbild. Die Darstellung der Inhalte kann auf verschiedene Weise erfolgen:

- Geometrische Modelle beschreiben die Form und räumliche Lage von Körpern.
- Topologische Modelle beziehen sich auf Lage und Anordnung von Körpern sowie deren Zusammenhang.
- Semantische Modelle stellen Eigenschaften jenseits von Geometrie und Topologie dar. Diese Eigenschaften sind nicht ableitbar sondern müssen vereinbart werden.
- Bei der feature – basierte Geometriemodellierung werden Geometriekörpern (=Features) Eigenschaften, Randbedingungen, Regeln etc. zugewiesen. Das Zusammenfügen von Features erzeugt Abhängigkeiten und Verbindungen zwischen Bauteilen.
- Objektorientierte Modelle stellen reale Sachverhalte dar. Objekte können Eigenschaften und Funktionen enthalten, die ihrem realen Vorbild entsprechen.

3.1.2.2 Modellarchitektur

Die Modellarchitektur bezeichnet die Strukturierung des Datenmodells und die Zusammenhänge zwischen den Teilmodellen. Ziel der Modellarchitektur ist ein größtmöglicher Datenaustausch bei geringem Implementierungsaufwand des Datenmodells.

Es werden verschiedenste Modellformen unterschieden. Zu den wichtigsten gehören das neutrale Modell, verschiedene Formen des Applikationsdomänenmodells und Produktmodelle, die ein Metamodell beinhalten (Hörenbaum, 2002).

Das neutrale Modell stellt den Idealfall dar. Jedes Anwendungsprogramm „versteht" das gesamte Datenmodell. Der Datenaustausch ist uneingeschränkt. Bei komplexen Modellen bedeutet dies einen zu großen Implementierungsaufwand.

Das Applikationsdomänenmodell ermöglicht den Datenaustausch innerhalb genau definierter Anwendungsgebiete (Domänen). Bestehen zwischen den Domänen keine Gemeinsamkeiten, kann kein Datenaustausch stattfinden. Der Implementierungs-aufwand hängt von der Größe der Domäne ab.

Besitzen Applikationsdomänenmodelle gemeinsame Ressourcen, kann durch die Nutzung dieser Ressourcen ein Datenaustausch zwischen den Domänen stattfinden. Die Ressourcen bilden ein Teilmodell und beinhalten grundlegende und für alle Domänen gültige Darstellungskonzepte (z. B. Maßeinheiten, Geometrie). Ein Datenaustausch auf niedrigem Niveau zwischen verschiedenen Domänen wird ermöglicht.

Ein gemeinsamer Modellkern kann grundlegende, allen Domänen gemeinsame Objekte definieren. Diese Objekte besitzen einen niedrigen Detaillierungsgrad und werden durch Vererbung in den Domänen spezialisiert. Daten aus der Spezialisierung gehen bei Austausch auf höherem Detaillierungsniveau zwischen den Domänen allerdings verloren.

Die höchst entwickelte Modellarchitektur besitzt ein Applikationsdomänenmodell mit gemeinsamem Kern und einem Metamodell. Im Metamodell sind generische Datenstrukturen definiert. Dadurch wird garantiert, dass überall im Modell dieselben Datenstrukturen verwendet werden. Durch den Einsatz des Metamodells ähneln die Domänen einander. Man spricht von einer Homogenisierung des Modells. Eine einfachere modulare Erweiterung und Implementierung wird ermöglicht. Dieser Modellart entsprechen die Industry Foundation Classes.

3.1.2.3 Unterschiede zu Geometriemodellen

Jede Softwareapplikation besitzt ein internes Datenmodell, das die Verarbeitung von Sachverhalten, ihren Eigenschaften und Beziehungen beschreibt. Klassische zwei- und dreidimensionale CAD – Anwendungen verwenden zur Beschreibung interner Daten Punkte, Linien, Rechtecke und Ebenen. Eigenschaften jenseits der Geometrie können Objekten nur in geringem Umfang zugeordnet werden, z. B. Linientyp.

Die folgenden Abbildungen sollen den Unterschied zwischen einem geometrischen Datenmodell und einem Gebäudedatenmodell verdeutlichen. Abbildung 3.2 stellt ein vereinfachtes Geometriemodell dar. Zwei Rechtecke bilden die Raumbegrenzungen. Die Eigenschaften der Rechtecke beziehen sich auf einen Ursprungspunkt, Länge und Breite. Der Punkt wiederum wird auf seine zweidimensionalen Koordinaten bezogen. Die Rechtecke werden nicht als Wände erkannt. Beziehungen bestehen zwischen den Rechtecken und den Punkten.

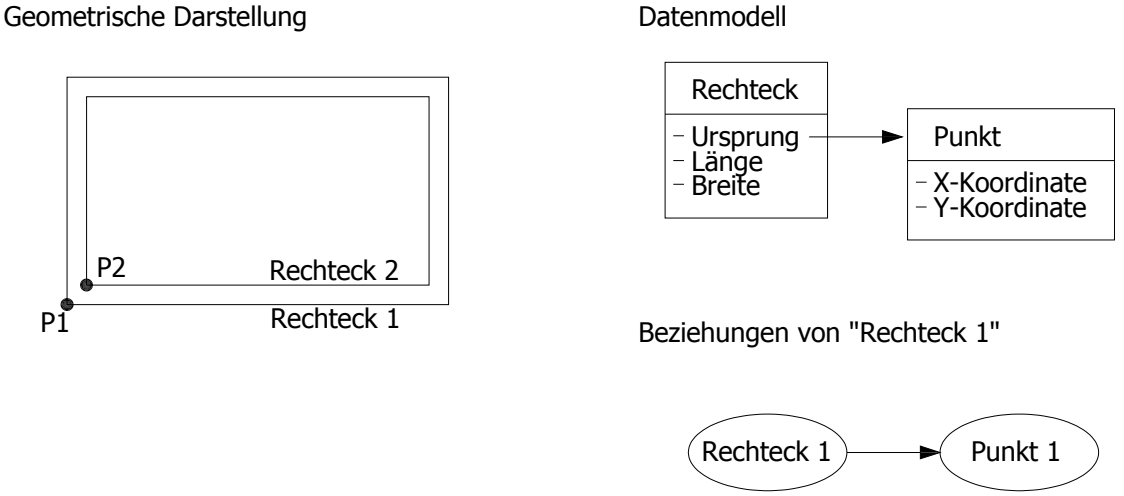


Abbildung 3.2: Geometrisches Datenmodell (nach Khemlani, 2004)

In Abbildung 3.3 ist ein stark vereinfachtes Gebäudemodell dargestellt. Auch hier besitzen die Wände Startpunkte, darüber hinaus werden zahlreiche weitere Eigenschaften zugewiesen, z. B. Länge, Höhe, Tiefe, Material und angrenzende Wände. Die Wände werden wiederum auf einen Raum, d. h. einen nicht – geometrischen Sachverhalt, bezogen. Das Gebäudemodell orientiert sich an Objekten, die Eigenschaften besitzen und untereinander in Beziehung stehen.

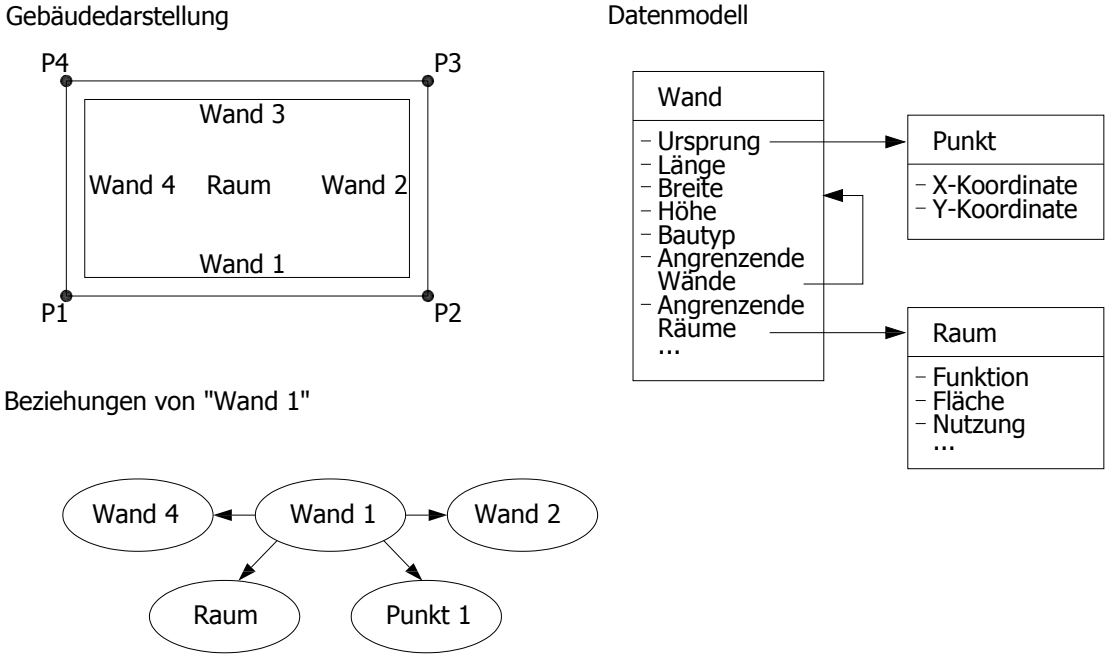


Abbildung 3.3: Gebäudedatenmodell (nach Khemlani, 2004)

Die Geometrie stellt nur einen Teil des Gebäudemodells dar. Darüber hinaus sind vor allem physisch nicht darstellbare Sachverhalte, wie Räume, von besonderer Bedeutung. Der Raum legt Beziehungen zwischen Wänden, Fenstern, Decken und Böden fest. Diese im Gebäudemodell bereits enthaltenen Daten fehlen dem geometrischen Modell und bedeuten einen zusätzlichen Arbeitsaufwand bei der Verarbeitung von reinen Geometriemodellen.

3.1.3 ISO 10303: STEP

In der internationalen Norm ISO 10303 ist ein Standard für den Datenaustausch zwischen CAD/CAM – Softwareanwendungen und Produktmanagementsystemen festgelegt. Die Norm wird meist als STEP bezeichnet – Standards for the Exchange of Product Model Data.

Schnittstellen für Softwareanwendungen werden erst durch Vereinbarungen zwischen Entwicklern möglich. STEP stellt einen weltweit anerkannten und allgemeingültigen Standard für Produktmodelle dar. Bereits gewonnene Erfahrungen über den Produktdatenaustausch in der Flugzeug-, Maschinen- und Automobil-industrie flossen in den STEP – Standard ein.

Die Norm beinhaltet Regeln und Methoden für Produktmodelle unabhängig ihres Anwendungsgebietes. Implementationsmethoden und eigenständige Produktmodelle

für bestimmte Anwendungen sind ebenfalls enthalten. Der Datenaustausch ist neutral und herstellerunabhängig.

Die wichtigsten Komponenten von STEP sind

- Methoden zur Datenmodellierung,
- Methoden zur Implementierung,
- Vorgaben für Konformitätsprüfungen,
- die Definition anwendungsunabhängiger Ressourcen und
- eigenständige Produktmodelle.

Unter Datenmodellierung wird die Beschreibung des Produkts verstanden. STEP verwendet dazu die Spezifikationssprache EXPRESS. Durch die formale Definition der Sprachelemente wird Eindeutigkeit und Maschinenlesbarkeit ermöglicht. EXPRESS ist keine Programmiersprache.

Zu den wichtigsten Sprachelementen gehören Entities, Attribute, Typen und Verweise. Ein Entity dient zur Abbildung eines Sachverhalts. Es enthält Attribute und Assoziationen zu anderen Entities. Attribute bilden die Eigenschaften von Entities ab.

Um die Übersichtlichkeit komplexer Produktmodelle zu erleichtern, wird die graphische Variante von EXPRESS verwendet. Mit EXPRESS – G werden Entities, ihre Attribute und Zusammenhänge in Schemata dargestellt.

In den folgenden Abbildungen wird ein einfaches Datenmodell in Text- und Grafikform dargestellt. Das Beispiel soll die Unterschiede in den Darstellungsmöglichkeiten zeigen und wurde Hörenbaum (2002) entnommen. Die genaue Definition der Syntax findet sich sowohl bei Hörenbaum, als auch in der ISO 10303-11 und im Implementation Guide der Industry Foundation Classes.

```
SCHEMA beispiel;

ENTITY bauteil
      SUPERTYPE OF
      (ONEOF (traeger, rechteck_wand));
        bezeichnung   : STRING;
        position      : STRING;
END_ENTITY;

ENTITY traeger
      SUBTYPE OF (bauteil);
        profil   : STRING;
        laenge   : REAL;
END_ENTITY;

ENTITY rechteck_wand
      SUBTYPE OF (bauteil);
        dicke    : REAL;
        laenge   : REAL;
        hoehe    : REAL;
END_ENTITY;

END_SCHEMA;
```

Abbildung 3.4: Definition einer einfachen Vererbungsstruktur in EXPRESS

Abbildung 3.4 zeigt ein einfaches Datenmodell, das die Vererbungsstruktur für Bauteile definiert. Die Eigenschaften (Bezeichnung und Positionsnummer) des Entities *bauteil* werden an die Untertypen vererbt. Die Untertypen *traeger* und *rechteck_wand* besitzen darüber hinaus andere Eigenschaften. Man spricht von einer Spezialisierung in Richtung der Vererbung.

Um die Struktur dieses einfachen Modells zu veranschaulichen, wird die graphische Variante EXPRESS – G verwendet. Die Vererbung wird hier durch eine dicke Linie dargestellt (Abbildung 3.5).

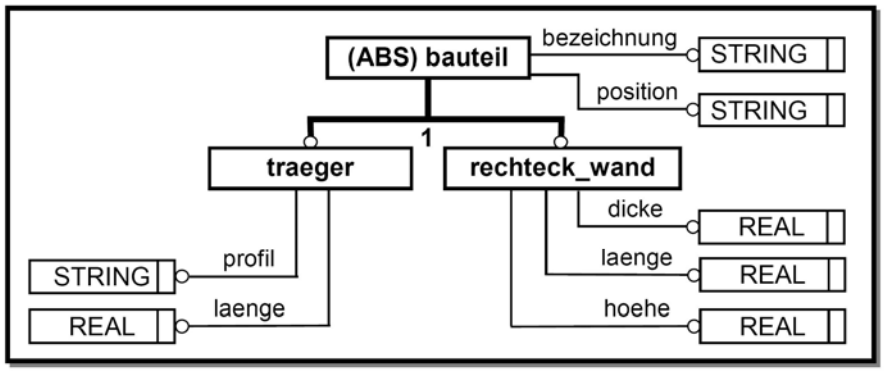


Abbildung 3.5: Darstellung des Beispiels in EXPRESS – G

Über die Datenmodellierung hinaus enthält STEP auch Methoden zur Implementierung. Als Implementierung wird die Einbindung eines Produktdatenmodells in Anwendungssoftware bezeichnet. STEP unterscheidet vier Stufen des Datenaustausches, die sogenannten *Implementation Levels*.

Die erste Stufe bildet der Datenaustausch über Austauschdateien (Step Physical File Exchange, SPF). Ein standardisiertes Zwischenformat, das Repository, stellt die zweite Stufe der Implementierung dar. Höchste Komplexität und Leistung bieten Datenbankmanagementsysteme und wissensbasierte Systeme. Die Implementierung erfolgt in allen Stufen unabhängig von der Definition des Produktdatenmodells.

Ein weiterer Teil des STEP – Standards sieht Konformitätsprüfungen an Hand abstrakter und konkreter Testfälle und Referenzbeispiele vor. Die Prüfung muss den formalen Vorgaben der Norm entsprechen und sieht genaue Kriterien für das Bestehen oder Nichtbestehen vor.

Die Nutzung gemeinsamer Ressourcen ist Grundlage des Produktdatenaustausches zwischen Teilmodellen (vgl. Abschnitt 3.1.2.2). ISO 10303 definiert anwendungsneutrale, gemeinsame Ressourcen, die in jedem Produktdatenmodell verwendet werden. Diese Ressourcen beinhalten geometrische, visuelle und topologische Darstellung, Materialeigenschaften, Toleranzen, Verwaltung von Prozessstrukturen, technische Zeichnungen und Kinematik. Ebenfalls definiert sind anwendungsabhängige Ressourcen, die in bestimmten Anwendungsgebieten unterstützt werden.

Neben den genannten Regeln und Methoden zur Definition und Anwendung von Produktdatenmodellen, bietet STEP auch vollständige Produktdatenmodelle für bestimmte Anwendungsgebiete. Sogenannte *Application Protocols* (APs) existieren u.

a. für allgemeine technische Zeichnungen, Regelungstechnik und Metallverarbeitung. Das AP 225 ist ein Produktmodell für das Bauwesen, das zur Beschreibung von Gebäudemodellen dient. Hörenbaum (2002) nennt die Art der Geometriedarstellung als einen der Gründe für den fehlenden Einsatz des AP 225 in kommerziellen CAD – Anwendungen.

Neben dem Produktdatenmodell der IFC, das auf STEP basiert, ist der Einsatz des Standards in der Bauindustrie zur Zeit gering. Für die Standardisierung des Datenaustausches muss erst ein Bewußtsein geschaffen werden. Die Auswirkungen des STEP – Standards eröffnen auch für die Bauindustrie neue Möglichkeiten im Projektablauf:

- Qualität, Genauigkeit und Vollständigkeit des Datenaustausches zwischen CAD – Programmen werden verbessert.
- Die Wiederverwendung bereits existierender Daten wird gefördert.
- Die Kombination von CAD – Daten mit semantischen Daten des Produkts wird vereinfacht.
- Die Erstellung einer gewerkübergreifenden Produktdatenbank wird erleichtert.

Ein Nachteil in der Entwicklung neuer Produktdatenmodelle im STEP – Standard sind die zeitaufwändigen ISO – Prozeduren. Die IFC orientieren sich teilweise am STEP – Standard, sind jedoch nicht an die ISO gebunden.

4 Das Produktdatenmodell der IFC

Dieser Teil der Arbeit widmet sich Zielen, Aufbau und Funktionen der IFC. Nach einer kurzen Übersicht der Datenstruktur, werden einzelne Teile der IFC detailliert betrachtet. Darüber hinaus werden Einsatzbereiche, Entwicklung und Implementierung des Modells besprochen. Methoden des Datenaustausches, die Anwendung der IFC und Erfahrungen aus Pilotprojekten bilden einen weiteren Schwerpunkt.

4.1 Ziel und Funktion

Die *Industry Foundation Classes* sind ein Produktdatenmodell, das speziell für die Baubranche entwickelt wurde. Produktdatenmodelle beinhalten alle ein Produkt bzw. Gebäude beschreibenden Daten. Sie ermöglichen die Kommunikation von Software gleicher oder verschiedener Anwendungsgebiete und dadurch die Wiederverwendung einmal erstellter Daten.

Kennzeichen der IFC ist die objektorientierte Modellierung des Gebäudes. Wände, Fenster oder Räume sind Objekte, die Eigenschaften besitzen. Wird eine IFC – Datei ausgetauscht, erkennt ein kompatibles Programm die Objekte und ihre Eigenschaften als solche. Im Gegensatz dazu übertragen klassische CAD – Anwendungen einzelne Punkte, Linien oder Blöcke ohne Eigenschaften jenseits der Geometrie zu berücksichtigen (vgl. Abschnitt 3.1.2.3).

Die IFC wurden von der IAI, der International Alliance for Interoperability, entwickelt. Die IAI ist eine unabhängige Organisation, die 1995 in den USA gegründet wurde. Weltweit bestehen nationale Gruppen, die sogenannten *Chapters*. Das deutsche Chapter stellt sich auf *www.buildingsmart.de* vor. Die IAI hat zur Zeit ca. 600 Mitglieder. Sie steht allen Unternehmen des Bau- und Baunebengewerbes, des Facility Managements sowie Forschungs- und Schulungseinrichtungen offen.

Ziel der IAI ist die kontinuierliche Entwicklung und Etablierung der IFC als herstellerunabhängiges Datenaustauschformat für Gebäudemodelle. Der Standard der IFC soll international unterstützt werden.

4.2 Einsatzbereiche

Ein Gebäudemodell im IFC – Format beinhaltet Daten verschiedenster Disziplinen. Es ist sowohl während der Entwurfs- als auch in der Bauphase von Nutzen. Im laufenden Betrieb kann das IFC – Gebäudemodell in Anwendungsprogramme des Facility Managements übernommen werden. Spätere Verwendungen des Modells hängen allerdings von der Qualität und dem Umfang des in der Entwurfsphase erstellten Modells ab.

In Abbildung 4.1 sind die Einsatzmöglichkeiten der IFC – Schnittstelle in der Entwurfsphase des Bauwerks dargestellt. Die ursprüngliche Erstellung des Modells geschieht in einem Architektur – CAD – Programm. Andere Planer können über ein zentrales Projektmanagementsystem auf das Modell zugreifen.

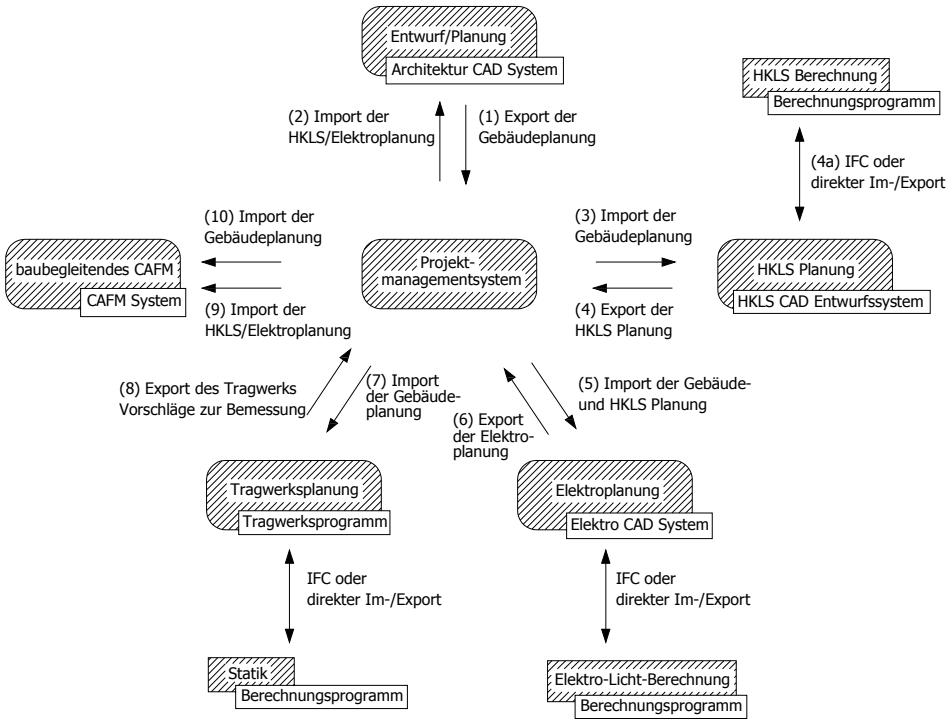


Abbildung 4.1: Einsatzmöglichkeiten der IFC- Schnittstelle in der Entwurfsphase (vereinfacht nach Liebich und Hoffeller, 2006)

Das CAAD – Programm exportiert das ursprüngliche Modell (1) und importiert Ergebnisse der Haustechnik und Elektroplanung in das Architekturmodell (2). HKLS – Planungsprogramme importieren das Architekturmodell als Grundlage ihrer Planung (3). Ergebnisse werden in das IFC – Modell übertragen (4). Greift das HKLS – Programm auf ein spezielles Berechnungsprogramm zu (z. B. thermische Gebäudesimulation) kann der Datenaustausch ebenfalls über das IFC – Modell erfolgen oder über eine programmeigene Schnittstelle (4a). Elektroplanung (5,6), Tragwerksplanung (7,8) und das baubegleitende Facility Management (9,10) importieren analog relevante Teile des IFC – Modells und stellen Berechnungsergebnisse über das Projektmanagementsystem zur Verfügung.

4.3 Datenstruktur der IFC

Das Datenmodell der IFC ist sehr umfangreich und komplex. In diesem Abschnitt werden nach einer Übersicht über das Modell, die Kernbereiche genauer betrachtet. Für eine eingehendere Beschreibung der Programmierung wird an dieser Stelle auf den *Model Implementation Guide* der IAI und die Online – Dokumentationen, die für die aktuellen Versionen erhältlich sind, verwiesen (Liebich, 2004).

4.3.1 Übersicht des Datenmodells

Das Produktdatenmodell der IFC basiert auf der Programmiersprache EXPRESS, die in der ISO 10303 definiert ist. Es besteht aus vier funktionalen Schichten, den sogenannten Layern. Die Layer sind modular aufgebaut, wodurch das Datenmodell einfacher zu erweitern ist. Abbildung 4.2 zeigt die hierarchische Struktur der Layer. Einzelne Module innerhalb der Layer der aktuellen IFC – Version sind in Abbildung 4.15 dargestellt (Abschnitt 4.5). Innerhalb dieser Hierarchie gelten strenge Vererbungsregeln. Eine Unterklasse darf nicht mehrere unmittelbare Oberklassen haben. Attribute, die von der Oberklasse geerbt werden, dürfen nicht umdefiniert werden. Die abstrakte Datenstruktur des Modells wird als generische Datenstruktur bezeichnet und ist im Modellkern (Kernel) festgelegt. Sie wird an die Klassen der höheren Layer weitervererbt und dort spezialisiert.

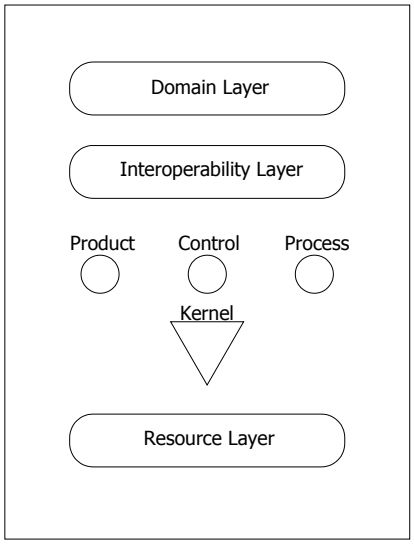


Abbildung 4.2: Aufbau der IFC Layer

Der *Resource Layer* bildet die unterste Hierarchieebene und enthält grundlegende Elemente wie Maßeinheiten, die von höheren Schichten verwendet werden können. Im *Core Layer* sind alle höheren Modellteile verankert. Er wird in *Kernel* und *Core Extensions* unterteilt und definiert alle Konzepte des Modells. Der *Interoperability Layer* bildet eine Zwischenschicht, auf die der Domain Layer aufbaut. In dieser Schicht sind Entities enthalten, die von den meisten Applikationen ausgetauscht werden, z.B. *Shared Building Elements* zum Austausch des architektonischen Designs.

Die oberste funktionale Schicht der IFC ist der *Domain Layer*. Er enthält Teilmodelle unterschiedlicher Anwendungsbereiche, z.B. FM, HVAC und Regelung. Dieser Layer bildet die höchste Vererbungsstufe der allgemeinen Strukturen des Kernels. Klassen der letzten beiden Schichten können instanziiert werden. Unter Instanziierung wird das Belegen mit konkreten Werten in einer Austauschdatei verstanden.

Das IFC – Modell basiert auf dem Top – Down – Ansatz. Ausgehend vom Bauwerk als Ganzem können die Bestandteile in mehreren Schritten in ihre Komponenten aufgelöst werden. Mit steigender Auflösung erhöht sich der Detaillierungsgrad und

die Datenmenge der Darstellung. Dieser Ansatz ermöglicht den Datenaustausch auf unterschiedlichen Detaillierungsniveaus. In den folgenden Abschnitten werden Ressourcen sowie generische und konkrete Datenstrukturen des Modells genauer betrachtet.

4.3.2 Ressourcen

Der unterste Layer der Hierarchie ist der Resource Layer. Hier ist die allgemeingültige, grundlegende Datenstruktur festgelegt, die von höheren Schichten genutzt werden kann. Diese Schicht beinhaltet gebäudeunabhängige Ressourcen wie Maßeinheiten, Geometrie oder Materialdefinition. In der Nomenklatur werden Entities dieser Schicht mit *Ifc…Resource* bezeichnet (z. B: IfcConstructionMaterialResource).

Attribute in höheren Layern verweisen auf Datentypen im Resource Layer. Dadurch wird z. B. die Maßeinheit nicht in der entsprechenden Klasse festgelegt, sondern einheitlich für alle Klassen im Resource Layer referenziert. Maßeinheiten können so projektspezifisch gewählt werden.

Für die Darstellung der Geometrie stehen geometrische Grundelemente sowie verschiedene Darstellungsarten für räumliche Oberflächen- und Volumenkörper-geometrien zur Verfügung (Hörenbaum, 2002). Die geometrische Darstellung darf nur geometrische Informationen beinhalten und muss vollständig sein. Objekte können mehrere geometrische Darstellungsarten besitzen, z. B. eine einfache und eine detaillierte Darstellung.

Die Materialdefinitionen erlauben neben der einfachen Bezeichnung von Materialien die Erstellung von detaillierten Schichtaufbauten von Bauteilen. Geschichtete Materialien sind in der Klasse *IfcMaterialLayerSet* definiert. Sie fasst mehrere Schichten (*IfcMaterialLayer*) zusammen.

4.3.3 Generische Datenstruktur

Im Core Layer befinden sich der Modellkern (Kernel) und seine Erweiterungen (Core Extensions). In dieser Schicht sind die generischen Strukturen des Modells festgelegt. Alle Klassen dieses Layers sind abstrakt, d.h. sie können nicht instanziiert (mit konkreten Werten) belegt werden. Die Instanziierung erfolgt durch Spezialisierung in den höheren Schichten.

Im Kernel sind die Schlüsselstrukturen des Datenmodells festgelegt. Er ist die Wurzel aller Datenstrukturen. Das IFC Modell besteht aus drei Schlüsselstrukturen: Objekt (IfcObject), Relation (IfcRelationship) und Eigenschaftsdefinition (IfcProperty-Definition). Diese drei fundamentalen Entities (Klassen) bilden die erste Ebene der Spezialisierung im IFC – Modell.

Die Schlüsselstrukturen und damit alle weiteren Objekte leiten sich von *IfcRoot* ab. IfcRoot ist die Oberklasse, d. h. die Wurzel aller Klassen des Core-, Interoperability- und Domainlayers. IfcRoot besitzt die Attribute *GlobalID* und *OwnerHistory*. Die

GlobalID beschreibt die Objektidentität, eine weltweit einzigartige Zeichenkette zur Wiederkennung eines Objekts. Diese Zeichenkette wird bei der Objekterstellung automatisch kreiert. Der Anwender nimmt keinen Einfluss darauf.

Die OwnerHistory klärt den Besitzstatus eines Objekts. Sie gibt Besitz- und Benutzerinformationen an. Die OwnerHistory des Projekts listet alle Änderungen im Modell chronologisch auf. Über einen IFC – Viewer (vgl. Kapitel 4.7.3) kann eine tabellarische Auflistung jener Anwender und Applikationen abgerufen werden, die Änderungen im Modell vorgenommen haben. Jedenfalls enthalten sind Angaben zum Benutzer, der verwendeten Applikation, Erstellungsdatum und Status. Zusätzlich können Einträge zur Art der Änderung und zum letzten ändernden Benutzer gespeichert sein.

Abbildung 4.3 zeigt den IfcRoot Subtype Tree, d. h. die Schlüsselstrukturen des IFC – Modells und ihre Ableitung von IfcRoot.

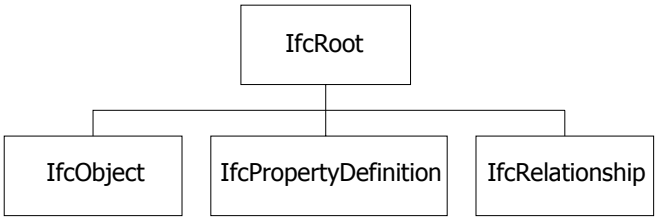


Abbildung 4.3: IfcRoot Subtype Tree

IfcObject bildet die abstrakte Oberklasse aller Objekte und beinhaltet greifbare Objekte und Gruppierungsmechanismen (Prozesse, Kontrollmechanismen…). IfcRelationship bildet die abstrakte Oberklasse aller Beziehungen zwischen Objekten. Im IFC – Modell werden Beziehungen zwischen Objekten als eigene Klassen dargestellt. Dieser Ansatz wird als *Objectified Relationships* bezeichnet.

Die abstrakte Oberklasse aller Eigenschaften wird von IfcPropertyDefinition gebildet. Semantische (nicht geometrische oder topologische) Eigenschaften werden nicht direkt als Attribute der Objekte sondern ebenfalls als eigenständige Klassen definiert. Es wird zwischen statischen und dynamischen Eigenschaften unterschieden. Statische Eigenschaften sind innerhalb des Datenmodells definiert. Sie sind unveränderlich und Unterklassen von IfcPropertyDefinition. Dynamische Eigenschaften sind außerhalb des Datenmodells definiert. Sie werden als dynamisch bezeichnet, da bei Bedarf Ergänzungen vorgenommen werden können. Für dynamische Eigenschaften steht die Klasse *IfcPropertySet* zur Verfügung. Diese sogenannten PSETs werden in der XML basierten Sprache PSD definiert. PSETs bieten einen Vorteil, wenn Informationen nur für wenige Beteiligte von Interesse sind oder sich nicht einheitlich darstellen lassen, z. B. wegen unterschiedlicher Normen.

Ebenfalls im Core Layer befindet sich die zweite Ebene der Spezialisierung des Modells. Jede der drei fundamentalen Schlüsselstrukturen (Objekt, Relation und Eigenschaftsdefinition) besitzt einen eigenen Vererbungsbaum (Entity Subtype Tree) zur Festlegung der Unterklassen (Entity Types). Die in dieser Schicht definierten

Klassen sind anwendungsunabhängig, d.h. sie beziehen sich auf kein spezielles Gewerk. Sie bilden die Grundlage der Core Extensions.

Abbildung 4.4 zeigt den IfcObject Subtype Tree. Beispiele für Unterklassen der Klasse IfcObject sind: *products*, *processes*, *controls*, *resources* und *project*. Zu den Unterklassen von IfcRelationship gehören: *assignment*, *association*, *definition* und *connectivity*. Von IfcPropertyDefinition leiten sich u.a. *type object* und *property set definition* ab.

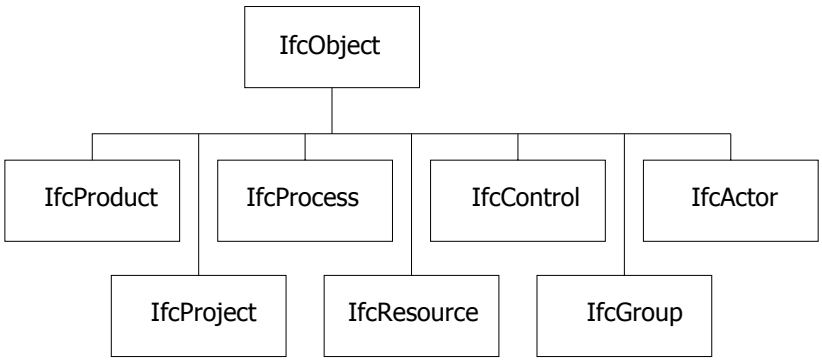


Abbildung 4.4: IfcObject Subtype Tree

Eine weitere Ebene der Spezialisierung in der IFC – Datenstruktur bilden die Core Extensions. Von der *Product Extension* leiten sich alle Unterklassen von IfcProduct, das im Kernel festgelegt ist, ab. In diesem Teil des Modells sind abstrakte Gebäudekomponenten wie Raum, Gebäude, Gebäudeelement oder Platz definiert. Die Spitze des Vererbungsbaumes für Randbedingungen und Zulassungen für jedes Objekt bildet die *Control Extension*. Sie definiert u.a. Zeitpläne und Informationen zum Lebenszyklus. In der *Process Extension* sind Oberklassen für Aufgaben, Arbeitsplanung, Prozesse u.v.a. festgelegt. Informationen über Prozesse werden wie Produkte angesehen und in eigenen Klassen ausgedrückt.

4.3.4 Konkrete Datenstruktur

Als konkrete Datenstrukturen werden Klassen bezeichnet, die in den Austauschdateien instanziiert werden (Hörenbaum, 2002). Diese Klassen werden auch *Leaf Node Classes* genannt, da sie am Ende eines Astes im Vererbungsbaum stehen. Sie werden im Gegensatz zu abstrakten Oberklassen in Austauschdateien geschrieben. Datensätze in diesen Dateien werden als Instanzen bezeichnet.

Im Interoperability Layer sind konkrete Strukturen enthalten, die den meisten Domain Anwendungen gemeinsam sind. Sie werden von den meisten Anwendungen ausgetauscht und sind wenig detailliert. Wichtigstes Modul des Interoperability Layers sind die *Shared Building Elements*, die tatsächliche Bauelemente enthalten.

Detaillierte, anwendungsspezifische Teilmodelle der IFC sind im Domain Layer enthalten. Sie werden nicht von allen Softwareanwendungen unterstützt. Ein Austausch dieser Teilmodelle kann nur innerhalb der Domäne erfolgen. Der Datenaustausch erfolgt auf hohem Detaillierungsgrad.

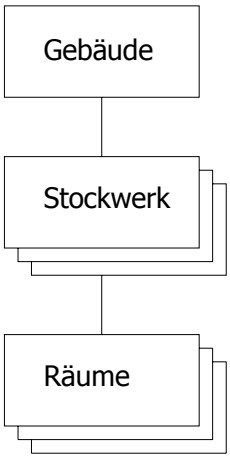


Abbildung 4.5: Raumordnung im IFC – Modell

Die wichtigsten, domänenübergreifenden Datenstrukturen des IFC – Modells sind die Projekt- und die Bauwerksstruktur.

Das Projekt stellt die größte Einheit in einer Austauschdatei dar. Es kommt genau einmal vor und ist der Rahmen für alle weiteren Daten. Die Klasse IfcProject verfügt über Attribute zur Beschreibung von Projektname und Projektphase. Über Verweise werden projektspezifische Angaben zu Maßeinheiten und geometrischer Darstellung festgelegt.

Abbildung 4.5 stellt die Bauwerksstruktur dar. Das Gebäude wird in die Bestandteile Stockwerk und Raum unterteilt. Diese nicht – gegenständlichen Elemente des Bauwerks können immer weiter bis zu ihren gegenständlichen Inhalten (Bauteilen) aufgelöst werden. Geschoßübergreifende Räume, Schächte und Treppenhäuser werden gegenwärtig nicht als ein zusammenhängender Raum betrachtet. Der betreffend Raum oder Schacht wird geschoßweise aufgeteilt. Die einzelnen Raumteile werden als eigenständiger Raum dem betreffenden Stockwerk zugeordnet.

4.3.5 Aufbau einer IFC – Datei

Für den Austausch von IFC – Gebäudemodellen wird das Datenaustauschformat der ISO 10303 benutzt, SPF (Step Physical File). Die IFC – Datei entspricht damit dem STEP – Standard und liegt in der Spezifikationssprache EXPRESS vor. Abbildung 4.6 zeigt den allgemeinen Aufbau einer STEP – Datei.

```
ISO-10303-21;

HEADER;
...
ENDSEC;

DATA;
...
ENDSEC;

END-ISO-10303-21;
```

Abbildung 4.6: Aufbau einer STEP Datei (nach Nour, 2005)

Der genaue Inhalt des IFC – Files wird in Textform dargestellt, wenn die Datei mit einem Editor geöffnet wird. In Abbildung 4.7 ist ein Auszug aus einem einfachen IFC – Gebäudemodell dargestellt. Die Datei besteht aus zwei Teilen. Der *Header* (Kopf) steht am Anfang der Datei und enthält Informationen über die verwendete IFC – Version. In diesem Beispiel wird das IFC2x2_FINAL Schema verwendet, das durch *FILE_SCHEMA (('IFC2X2_FINAL'))* gekennzeichnet ist. Ebenfalls enthalten sind Informationen über den Ersteller der Datei und der verwendeten Software. Im zweiten Teil der Textdatei, mit dem Wort *DATA* eingeleitet und mit *ENDSEC* beendet, befinden sich die eigentlichen Modelldaten. In jeder Zeile wird mit fortlaufender Nummerierung eine Instanz (Entity) des Modells aufgeführt. Attribute der Instanzen und Verweise sind ebenfalls aufgeführt. Nicht dargestellt

werden z. B. abstrakte Klassen. Inhalte des IFC – Modells können über den Editor geändert werden. Objekte können, ohne das Modell zu zerstören, gelöscht oder Angaben zu Personen und Anwendungen, die in der OwnerHistory enthalten sind, geändert werden.

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('IFC2X2_FINAL'),'2;1');
FILE_NAME('IfcTest.ifc','2006-04-05T12:33:30',('btm02031'), ('Data Design System ASA'),
FILE_SCHEMA(('IFC2X2_FINAL'));
ENDSEC;
DATA;
#1=IFCPERSON($,'\X\01-',$,$,$,$,$,$);
#2=IFCORGANIZATION($,'-',$,$,$);
#3=IFCPERSONANDORGANIZATION(#1,#2,$);
#4=IFCAPPLICATION(#2,'Version 6.35 Prerel build 24/10-2005','Building Services Partner
#5=IFCOWNERHISTORY(#3,#4,$,.NOCHANGE.,$,$,$,1144151098);
#6=IFCSIUNIT(*,.LENGTHUNIT.,$,.METRE.);
#7=IFCSIUNIT(*,.AREAUNIT.,$,.SQUARE_METRE.);
#8=IFCSIUNIT(*,.MASSUNIT.,$,.GRAM.);
#9=IFCSIUNIT(*,.ELECTRICCURRENTUNIT.,$,.AMPERE.);
#10=IFCSIUNIT(*,.ELECTRICVOLTAGEUNIT.,$,.VOLT.);
#11=IFCSIUNIT(*,.ELECTRICRESISTANCEUNIT.,$,.OHM.);
#12=IFCSIUNIT(*,.POWERUNIT.,$,.WATT.);
#13=IFCSIUNIT(*,.MASSUNIT.,$,.GRAM.);
#14=IFCSIUNIT(*,.PLANEANGLEUNIT.,$,.RADIAN.);
#15=IFCSIUNIT(*,.SOLIDANGLEUNIT.,$,.STERADIAN.);
#16=IFCSIUNIT(*,.THERMODYNAMICTEMPERATUREUNIT.,$,.DEGREE_CELSIUS.);
#17=IFCSIUNIT(*,.PRESSUREUNIT.,$,.PASCAL.);
#18=IFCSIUNIT(*,.LUMINOUSFLUXUNIT.,$,.LUMEN.);
#19=IFCSIUNIT(*,.ILLUMINANCEUNIT.,$,.LUX.);
#20=IFCSIUNIT(*,.VOLUMEUNIT.,$,.CUBIC_METRE.);
#21=IFCSIUNIT(*,.TIMEUNIT.,$,.SECOND.);
#22=IFCDERIVEDUNITELEMENT(#21,-1);
#23=IFCDERIVEDUNITELEMENT(#20,1);
```

Abbildung 4.7: Beispiel einer IFC – Datei (Auszug)

Anschaulicher kann der Aufbau der IFC – Datei über einen Vererbungsbaum dargestellt werden. Wird die Datei mit einem IFC – Viewer geöffnet, kann die Hierarchie übersichtlich dargestellt werden. Die fortlaufende Nummerierung der Instanzen wird dabei übernommen. Abbildung 4.8 zeigt den Vererbungsbaum eines einfachen IFC – Modells, der mit dem Programm IfcStoreyView dargestellt wurde. Softwareprogramme zur Analyse werden in Abschnitt 4.7.3 vorgestellt.

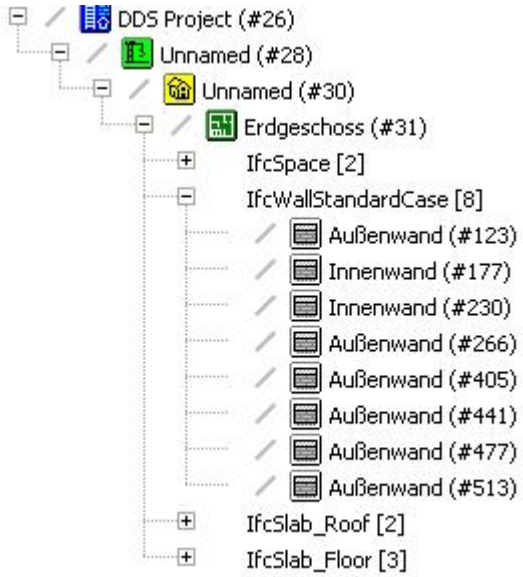


Abbildung 4.8: IFC Vererbungsbaum

Das Projekt „DDS Project (#26)" beinhaltet in diesem Beispiel das Grundstück „Unnamed (#28)". Das Gebäude „Unnamed (#30)" auf diesem Grundstück besteht aus einem Stockwerk, „Erdgeschoß (#31)". Das Erdgeschoß wiederum enthält zwei Räume (IfcSpace) sowie Wände (IfcWallStandardCase), Decken (IfcSlab_Roof) und Böden (IfcSlab_Floor). In eckigen Klammern ist die Anzahl der enthaltenen Instanzen dargestellt. Die fortlaufende Nummer wird mit # bezeichnet.

4.3.6 ifcXML

ifcXML ist die XML – Version der IFC und eignet sich besonders für den Austausch von semantischen Teilbereichen des IFC – Produktmodells. ifcXML – Dateien werden durch die Konvertierung eines bestehenden EXPRESS Schemas generiert. Die extensible markup language (XML) ist ein internationaler Standard für den Austausch elektronischer Dokumente. XML ist ein Textformat und wurde für die Anwendung im Internet optimiert. Entwickelt vom W3C (World Wide Web Consortium) ist XML eine lizenzfreie Metasprache, die dazu dient Datenstrukturen eindeutig zu halten (Bos, 1999).

Als Metasprache ermöglicht XML, Formate für Dokumente in sogenannten Dokumenttyp – Definitionen (DTD) festzulegen. Diese Formate können instanziiert werden. Man spricht von *Tagged Documents*. Abbildung 4.9 zeigt die Dokumenttyp – Definition in XML des einfachen Datenmodells aus Abschnitt 3.1.3.

```
<?xml version="1.0"?>
<xml_beispiel>

  <!ATTLIST traeger
            id           ID     #REQUIRED
            bezeichnung  CDATA  #REQUIRED
            position     CDATA  #REQUIRED
            profil       CDATA  #REQUIRED
            laenge       CDATA  #REQUIRED>

  <!ATTLIST rechteck_wand
            id           ID     #REQUIRED
            bezeichnung  CDATA  #REQUIRED
            position     CDATA  #REQUIRED
            dicke        CDATA  #REQUIRED
            laenge       CDATA  #REQUIRED
            hoehe        CDATA  #REQUIRED>

</xml_beispiel>
```

Abbildung 4.9: Dokumenttyp – Definition in XML (Hörenbaum, 2002)

In Abbildung 4.10 ist die instanziierte Austauschdatei des selben Datenmodells in XML dargestellt. Bereits bei diesem Beispiel zeigt sich, dass das Datenmodell im XML – Format größer als im STEP – Format ist, das EXPRESS verwendet. Weiters ist zu erkennen, dass XML keine Entsprechung der Vererbung kennt. Die Eigenschaften des ursprünglichen Entities *bauteil* wurden in den Untertypen *traeger* und *rechteck_wand* direkt definiert. Eine Lösung bietet der weiterführende Standard *XML – Schema*. Die Dokumenttyp – Definition wird durch eine *XML – Schema – Definition* (XSD) ersetzt. XSD bietet Erweiterungen um Datentypen und strukturelle Konzepte.

```
<?xml version="1.0"?>
<xml_beispiel>

  <traeger>
    <id>_1</id>
    <bezeichnung>Deckentraeger</bezeichnung>
    <position>POS1</position>
    <profil>HE300A</profil>
    <laenge>7500.0</laenge>
  </traeger>

  <rechteck_wand>
    <id>_2</id>
    <bezeichnung>Seitenwand</bezeichnung>
    <position>POS2</position>
    <dicke>250.0</dicke>
    <laenge>15000.0</laenge>
    <hoehe>2750.0</hoehe>
  </rechteck_wand>

</xml_beispiel>
```

Abbildung 4.10: Austauschdatei als XML – Dokument (Hörenbaum, 2002)

XML bildet auch für die Produktmodellierung eine Lösung. Seitens der Software – Industrie sind für diesen Standard viele kostengünstige Programmierwerkzeuge vorhanden. Es wird von Datenbanken unterstützt und ist Grundlage vieler eCommerce Applikationen und Internetservices.

Für die Konvertierung des EXPRESS – Schemas in XML stehen sogenannte *ifcXML Methodologies* zur Verfügung. Die angewandte Methode hängt von der IFC – Version des vorliegenden Gebäudemodells ab (Nisbet und Liebich, 2005). Die Übertragung (Mapping) des EXPRESS Schemas in ein XML – Schema ist in Teil 28 der ISO 10303 festgelegt. ifcXML – Dateien besitzen die Dateinamenerweiterung *.xml oder *.ifx.

Auf Grund der Definition von XML gehen beim Mapping des EXPRESS Schemas Informationen verloren. Zur Darstellung eines kompletten Gebäudemodells und vor allem seiner Geometrie eignet sich ifcXML nicht. Für den Datenaustausch innerhalb der AEC/FM – Industrie wird weiterhin das für die Produktdatenmodellierung entwickelte STEP – Format verwendet. Physische Austauschdateien im STEP – Format (SPF – Step Physical File) besitzen eine wesentlich geringere Dateigröße.

Trotz des Datenverlusts und der erheblich umfangreicheren Dateigröße gibt es in der Bauindustrie viele Einsatzbereiche für ifcXML. Vor allem dem Informationsaustausch mit weniger spezialisierten Anwendungen genügt die Übertragung von Teilmodellen, Berichten, Zeitplänen oder Bibliotheken. Die Dateigröße spielt hier eine geringere Rolle und XML ermöglicht die Kompatibilität mit Datenbanken, die meist XML – fähig sind.

Anwendungsgebiete von ifcXML sind Übertragungen zwischen dem IFC Produktdatenmodell und

- Dokumentbasierten Anwendungen (Zeitpläne, Produktdatenblätter, Materialbibliotheken),
- Nachrichtenorientierten Anwendungen (Bestellungen, eCommerce – Anwendungen),

- Kommunikation mit XML – basierten Produktmodellen, wie dem GIS – Objektmodell (Nisbet und Liebich, 2005).

4.3.7 Bewertung des IFC – Modells

In der Bewertung des IFC – Produktdatenmodells hebt Hörenbaum (2002) besonders die sehr leistungsfähigen, geometrischen Darstellungsmöglichkeiten hervor. An den Ansprüchen der IAI gemessen, befinden sich die IFC jedoch erst im Anfangsstadium. Besonders der geringe Detaillierungsgrad des Modell stehe der Praxis noch im Wege. Ebenfalls hindernd sind der große Aufwand zur Einarbeitung und Implementierung des Modells. Die Orientierung an der ISO 10303 bewertet Hörenbaum positiv.

Die wichtigsten Stärken der IFC sind:

- Internationalität
- Akzeptanz durch Software – Hersteller
- Zertifizierung kompatibler Software durch die IAI
- Modulare Erweiterbarkeit
- Flexibilität
- Einsatz der XML – Technologie
- Laufende Weiterentwicklung

Zukünftig dürften sich die IFC daher als das wichtigste Produktdatenmodell der Baubranche etablieren.

4.4 Implementierung

Die Implementierung, d. h. die Einbindung in Applikationssoftware, ist der wichtigste Schritt für den Einsatz eines Produktdatenmodells. Voraussetzung für die Implementierung ist ein stabiles und robustes Produktdatenmodell. Diese Eigenschaften des Modells können allerdings nur durch den Einsatz in entsprechenden Softwareanwendungen getestet und verbessert werden. Der Einsatz des IFC – Modells in realen Pilotprojekten ist zu seiner Entwicklung unumgänglich (siehe Kapitel 4.7.4).

Ein Problem der Implementierung ist, dass Softwareendnutzer die Vorteile von Interoperabilität noch nicht kennen. Softwareentwickler müssen daher erst den umfangreichen Aufwand für die Implementierung des Produktdatenmodells in Kauf nehmen, um das Modell in seiner Anwendung verbreiten zu können.

Das IFC – Modell wurde speziell für den Einsatz in der Baubranche entwickelt. Einzelne Disziplinen innerhalb der Branchen grenzen sich stark voneinander ab. Ihre Softwareanwendungen bilden sogenannte Insellösungen. Die Datenübertragung zwischen Softwareanwendungen verschiedener Disziplinen ist meist nicht möglich.

Jede Software besitzt ein eigenes internes Datenmodell, das genau auf die Aufgaben des Programms abgestimmt ist. Die Implementierung des IFC – Modells basiert auf

dem Mapping interner Daten in das IFC – Modell. Unter Mapping wird das Übertragen von Daten aus einem Format in ein anderes verstanden.

Diese Übertragung interner Daten in das austauschbare IFC – Modell erfordert einen großen Programmieraufwand. Es muss sichergestellt werden, dass das Programm die Daten korrekt interpretiert und vollständig in ein IFC – Modell überträgt. Weiters muss ein Mapping in zwei Richtungen möglich sein, um Daten sowohl in ein IFC – Modell exportieren als auch aus einem bestehenden Modell importieren zu können. Das Mapping interner Daten kann durch interne Programmteile oder externe Module bewerkstelltigt werden.

4.4.1 Verschiedene Arten von IFC – Kompatibilität

Die Implementierung einer IFC – Schnittstelle bedeutet, dass die Software-anwendung das programminterne Datenmodell in das IFC – Produktmodell übersetzt. Für die Datenübertragung stehen mehrere Wege zur Verfügung (Abbildung 4.11).

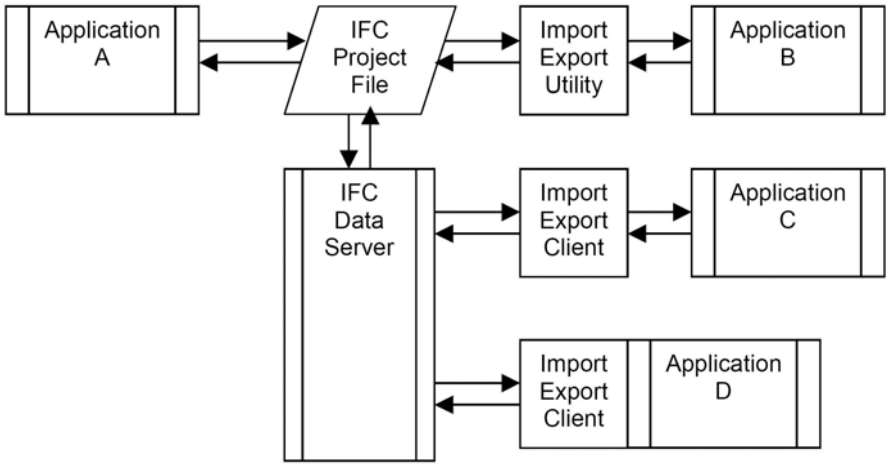


Abbildung 4.11: IFC Kompatibilität (Hitchcock, 2002)

Applikation A besitzt eine integrierte Schnittstelle, um direkt auf IFC – Dateien zugreifen zu können. Applikation B wird der Zugriff auf das Modell über eine externe Schnittstelle ermöglicht. Beide Varianten greifen auf eine ASCII – Textdatei zu, die das Modell im EXPRESS Standard enthält. Der Einsatz von Servern ermöglicht eine alternative Zugriffsmöglichkeit auf das IFC – Modell. Das Gebäudemodell wird auf dem Server gespeichert. Anwendungen greifen über ein Interface indirekt auf IFC – Daten zu. Clients (Benutzer) des Servers besitzen entweder eine externe (Applikation C) oder interne Schnittstelle (Applikation D). Der Einsatz von Modell Servern bietet den Vorteil, dass sich Änderungen an der Verarbeitung des Modells durch den Server nicht direkt auf die Entwicklung der zugreifenden Software auswirkt.

4.4.2 Methoden des Datenaustausches

Die gewählte Form des Datenaustausches hängt von der weiteren Verwendung des Gebäudemodells ab. Für den Datenaustausch spielt sowohl das Austauschformat als auch der Datenträger eine Rolle. Abbildung 4.12 zeigt den Zusammenhang zwischen Austauschformaten, Datenträgern und dem Gebäudemodell.

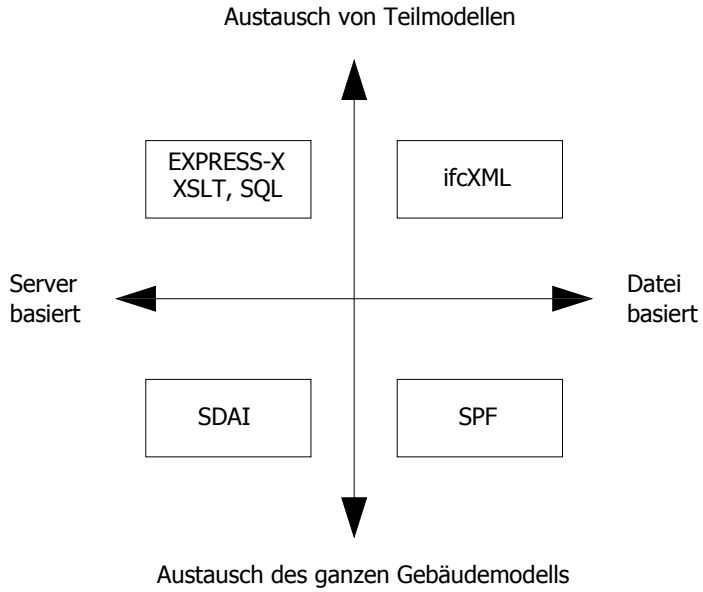


Abbildung 4.12: Möglichkeiten des Datenaustausches über das IFC - Modell (nach Nisbet und Liebich, 2005)

Für den Austausch von Teilmodellen eignet sich besonders die XML – Form der IFC (*ifcXML*). Ausgetauscht werden hauptsächlich nicht – geometrische Daten. Die erhöhte Dateigröße im XML – Format spielt in diesem Fall eine geringe Rolle. Sollen Teile eines Gebäudemodells ausgetauscht werden, das auf einem Server gespeichert ist, eignen sich datenbankorientierte Austauschformate wie *SQL* oder *EXPRESS – X*.
Um das gesamte Gebäudemodell auszutauschen, wird für den dateiorientierten Datenaustausch das *SPF* – Format des STEP – Standards verwendet. Komplexe Gebäudemodelle beanspruchen erhebliche Speicherkapazitäten. Sie können mit Hilfe von Servern einfacher gespeichert, verwaltet und ausgetauscht werden. Der STEP – Standard bietet für den Austausch von Informationen über eine Datenbank eine Schnittstelle an. Das *SDAI* (Standard Data Access Interface) ermöglicht den Datenbankzugriff. SDAI ist eine in ISO 10303 standardisierte Programmierschnittstelle zur Implementierung von Produktdatenmodellen.

4.4.3 IFC – Modell Server

Beim Austausch von IFC – Gebäudemodellen mittels Dateien treten drei Hauptprobleme auf (Kiviniemi et al., 2005):

1. Der Informationsbedarf von Anwendungen unterscheidet sich stark, sodass es unmöglich ist, alle Daten in einem Modell zu verwalten, das zwischen den Anwendungen ausgetauscht wird.
2. Instanziierte IFC – Gebäudemodelle sind sehr groß. Der Datenaustausch wird dadurch erschwert. Zudem würden die meisten Programme nur Teilmodelle benötigen.
3. Werden Austauschdateien verwendet, ist die Kontrolle von Zugriffs- und Bearbeitungsrechten sowie der Versionierung der Daten nicht möglich.

2001 begann die Entwicklung von sogenannten IFC – Modell Servern. Derzeit sind zumindest drei IFC – kompatible Modell Server erhältlich. Der Server speichert alle Daten des Gebäudemodells in einer zentralen Datenbank. Die Anwender sind Benutzer (Clients) des Servers und greifen auf Teilmodelle zu.

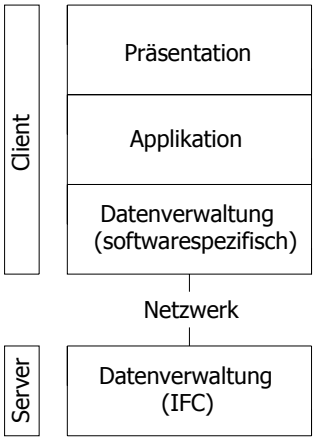


Abbildung 4.13: Geteiltes Datenverwaltungssystem (nach Korpowski, 2003)

Beim Import der IFC – Daten werden diese in anwendungsspezifische Datensätze umgewandelt. Die Datenverwaltung wird von Server und Client durchgeführt. Abbildung 4.13 zeigt die Funktionen von Server und Client in einem geteilten Datenverwaltungssystem.

Der Server ermöglicht die konsitente Datenhaltung während des gesamten Produktlebenszyklus. Die Daten werden nicht redundant gespeichert. Das ursprüngliche Gebäudemodell wird vom Ersteller auf dem Server zur Verfügung gestellt. Zur Abfrage von Daten aus der Server – Datenbank wird eine Client – Software verwendet. Abhängig vom Fachbereich des Planers, seiner Zugriffsberechtigung und der benötigten Modelldaten erfolgt ein selektiver Import in die Anwendersoftware. Nach der Bearbeitung werden das Teilmodell und relevante Ergebnisse an den Server zurückgegeben. Bei der Integration der weiterbearbeiteten Teilmodelle in das Gesamtmodell auf dem Server ist eine Versionierung durchzuführen. Das Teilmodell überschreibt bestehende Datensätze in der Datenbank nicht. Die Benutzer bearbeiten eine neue Variante des Gesamtmodells. Vorteile der Abfrage von Teilmodellen sind die verringerte Größe des Datensatzes und die verkürzte Übertragungszeit. Das Einlesen in der Anwendersoftware wird erleichtert (Korpowski, 2003).

Modell Server eignen sich besonders für die Verwaltung von komplexen Gebäudemodellen und den Austausch von Teilmodellen. Erfahrungen mit Modell Servern in einem Pilotprojekt werden in Abschnitt 4.7.4.2 besprochen.

4.4.4 Middleware Tool: BS Pro COM – Server

Der IFC – Standard soll die Interoperabilität zwischen Software – Programmen im Bau- und Baunebengewerbe auf Basis eines herstellerunabhängigen Datenaustauschformats gewährleisten. Die Struktur der IFC ist komplex und sowohl während des Programmierens als auch in der Datenspeicherung sehr umfangreich. Die Integration des IFC – Standards in Softwareprogramme ist für Entwickler mit hohem Aufwand verbunden. Olof Granlund Oy hat ein „Vermittlungswerkzeug", den BSPro COM – Server, geschaffen, um Entwicklern einen einfachen Zugang zu IFC – Daten zu gewährleisten, ohne tieferes Verständnis für die Programmierung von IFC – Schnittstellen erwerben zu müssen.

Der BSPro COM – Server ist speziell für den Haustechnikbereich (*Building Services* – BS) entwickelt worden. Die meisten führenden Architektur – CAD – Programme (z. B. ArchiCAD, Autodesk Architectural Desktop, Nemetschek) sind IFC – kompatibel und können dem weiteren Bauprozess ein fertiges Gebäudemodell zur Verfügung stellen. Die Weiterverarbeitung der IFC – Daten (z. B. das Auslesen in Haustechnikanwendungen, das Aktualisieren oder Hinzufügen von Daten) ist allerdings komplex und fehleranfällig. Um ein erneutes, händisches Einlesen der Daten zu verhindern, vermittelt BSPro COM – Server zwischen der vorhandenen IFC – Datei und dem Anwendungsprogramm.

In Abbildung 4.14 wird der Einsatz von BSPro COM – Server im Datenaustausch dargestellt. Anwendungen während des gesamten Gebäudelebenszyklus (Simulation, Planung, Facility Management etc.) können einfach IFC – Kompatibilität erreichen.

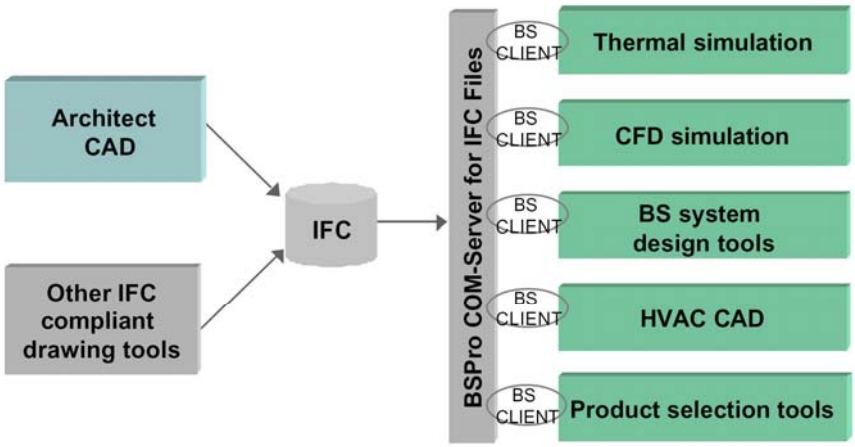


Abbildung 4.14: BSPro COM – Server im Datenfluss (Karola et al., 2001)

Der BSPro COM – Server besitzt einen sprachunabhängigen Aufbau und basiert auf der Microsoft COM Technologie. Softwareentwicklern wird die Möglichkeit geboten, durch den Einsatz von BSPro COM – Server mit geringem Aufwand IFC – kompatible Anwendungsprogramme zu realisieren. Zur Zeit werden die Übertragung der Geometrie des Gebäudemodells sowie Angaben zum thermischen Verhalten der Gebäudehülle in den IFC – Formaten 1.5.1, 2.0 und 2x unterstützt.

Neben dem Import von Geometriedaten können auch Daten in die IFC – Datei geschrieben und in andere Programme zur Weiterverwendung exportiert werden. Erwähnt sei hier die Verwendung von Simulationsergebnissen für die Dimensionierung von haustechnischen Anlagen.

Weiterentwicklungen im Einsatzbereich sind im auf Geometrie beschränkten Datenaustausch und in umfassenderen Angaben zu Wandoberflächen (wie z. B. von EnergyPlus benötigt) zu suchen.

Zur Zeit wird der BSPro COM – Server u. a. in folgenden Softwareentwicklungsprojekten eingesetzt (Olof Granlund Oy, 2006):

- Progman Oy: MagiCAD HVAC-CAD
- Ansys CFX: CFX software
- Massachusetts Institute of Technology (MIT): MIT CFD – Software
- Lawrence Berkeley National Laboratories (LBNL): EnergyPlus Energy Software (IFCtoIDF – Tool)

4.4.5 Implementation Views

1999 wurde von einer Gruppe unabhängiger Softwarehersteller, Softwarenutzer und Forschungsinstitute die BLIS – Initiative gegründet. BLIS steht für Building Lifecycle Interoperable Software. Ziel der Initiative ist es, die Implementierung und Nutzung des IFC – Modells zu unterstützen (BLIS Project, 2003).

Die Implementation des gesamten IFC – Produktdatenmodells würde den Umfang jeder Softwareanwendung bei weitem übersteigen. Applikationen nutzen den Teilbereich des Modells, der ihrem Anwendungsbereich entspricht. Diese Teilbereiche des Modells sind in sogenannten *Views* oder Sichten beschrieben.

Im Zuge des BLIS – Projekts wurden spezielle Austauschszenarien entwickelt, die in die IFC übernommen wurden. Diese *Implementation Views* ermöglichen die sinnvolle Verwendung der IFC in den definierten Szenarien, indem sie Anforderungen an Anwendungen sowie an die im IFC – Modell enthaltenen Objekte definieren. Insgesamt wurden sechs Views entwickelt:

- Design ⇔ Design (Geometry View)
- Client Briefing/Space Planning ⇔ Architectural Design
- Architectural Design ⇔ HVAC Design
- Arch/HVAC Design ⇔ Quantities Take Off / Cost Estimating
- Arch/HVAC Design ⇔ Thermal Load Calculations / HVAC System Design
- Arch/HVAC Design ⇔ Construction Management / Scheduling

Softwareanwendungen, die Implementation Views des BLIS – Projekts unterstützen sind u. a. ArchiCAD, Solibri Model Checker und RIUSKA.

4.5 Bisheriger Entwicklungsverlauf

Die erste Version der IFC wurde 1997 veröffentlicht. IFC1.0 enthielt grundlegende Teile des IFC – Datenmodells und wurde von den Versionen IFC1.5.1 (1998) und IFC2.0 (1999) abgelöst. 1.5.1 war die erste Version, die in kommerzielle Softwareanwendungen implementiert wurde.

Im Oktober 2000 wurde die Version IFC2x veröffentlicht. Diese Version der IFC stellt eine für einige Jahre unveränderliche Plattform des Modells dar. Auf dieser Basis können anwendungsspezifische Komponenten der IFC weiterentwickelt werden. Die Spezifikationen der IFC2x Plattform wurden 2005 in der ISO/PAS 16739 standardisiert. Die nachfolgenden Versionen sind Erweiterungen der IFC2x Plattform. Die aktuellste Version, IFC 2x Edition 3 (kurz IFC2x3) wurde Anfang 2006 veröffentlicht. Die aktuelle Spezifikation für ifcXML liegt für IFC2x2 Addendum 1 vor. Die Layerstruktur von IFC2x3 ist in Abbildung 4.15 dargestellt.

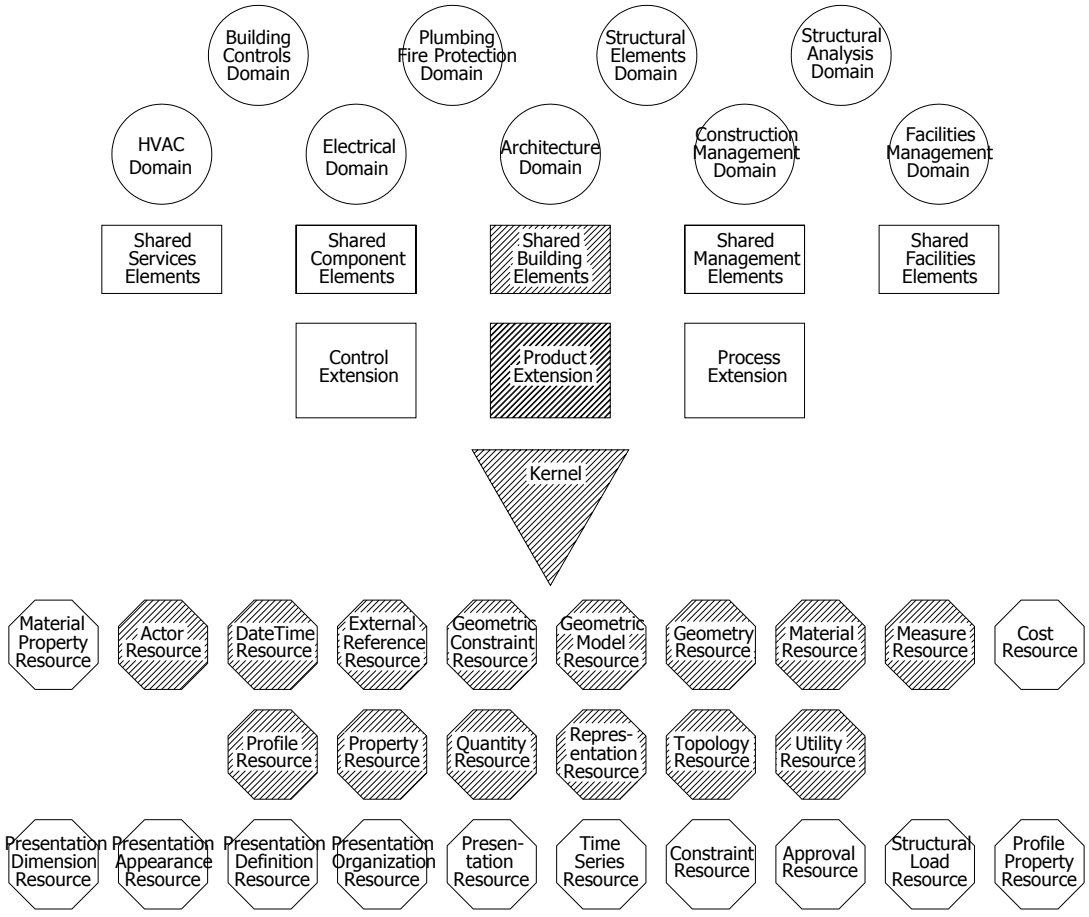


Abbildung 4.15: Schema des IFC2x3 Modells (nach IAI, 2006b)

Module der ISO – zertifizierten IFC2x Plattform sind schraffiert. Achteckig dargestellte Module sind Teil des Resource Layers. Kernel und Control-, Product- sowie Process Extension bilden den Core Layer. Die darüberliegenden, rechteckig dargestellten Module sind Teil des Interoperabilitiy Layers. Einzelne Anwendungsgebiete des Domain Layers sind kreisförmig dargestellt.

4.6 Weiterentwicklung des Modells

4.6.1 Extension Projects

Die Weiterentwicklung der IFC erfolgt durch Erweiterungen des Modells oder durch langfristig sinnvolle Änderungen des Modells selbst. Beide Entwicklungsformen sollen in einigen Jahren schließlich zur Einführung einer neuen, verbesserten IFC Plattform führen (IAI, 2006a).

Extension Models sind Erweiterungen des IFC – Modells und basieren auf der IFC2x Plattform. Die Erweiterungen werden von unabhängigen, eigenfinanzierten Projektgruppen entwickelt und von der IAI begleitet und überprüft. Eine Übersicht der aktuellen Erweiterungsprojekte sowie Vorschläge für neue Projekte sind auf der IAI – Homepage zu finden.

4.6.2 Zugriff auf IFC – Modell Server mit SABLE

Softwareanwendungen können über eine API - Programmierschnittstelle (Application Programming Interface) auf ein IFC – Gebäudemodell auf einem Server zugreifen. Diese Schnittstellen sind für jede Anwendung individuell gestaltet. Um mit den Anwendungen kommunizieren zu können, müssen mehrere Interfaces im Server implementiert sein (Abbildung 4.16).

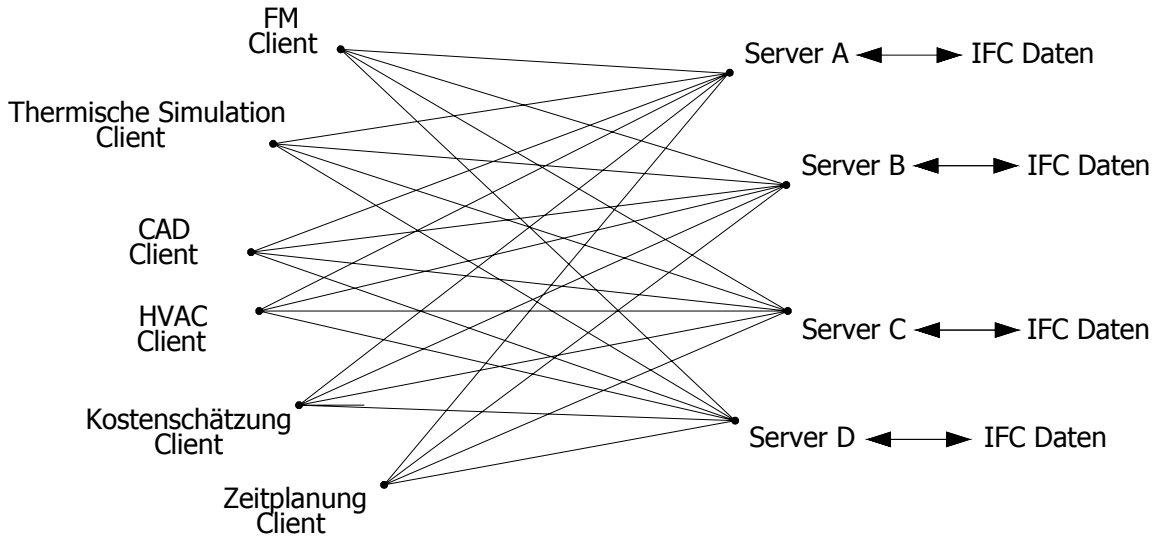


Abbildung 4.16: Serverzugriff ohne standardisiertes Interface (vereinfacht nach BLIS Project, 2003)

Ziel der Entwickler von SABLE (Simple Access to the Building Lifecycle Exchange) ist es, ein standardisiertes Interface zu schaffen (Abbildung 4.17). Eine Harmonisierung des Serverzugriffs wird durch die Definition von einfachen API für IFC – Modell Server und hoch spezialisierte API für einzelne Domänen der IFC erreicht (BLIS Project, 2003).

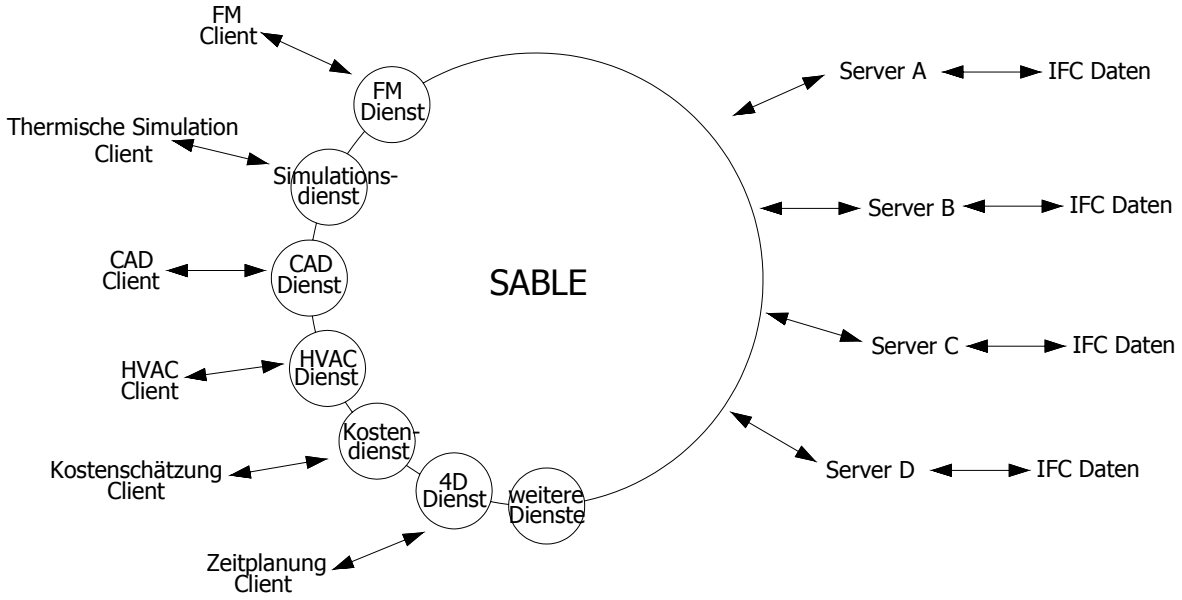


Abbildung 4.17: Serverzugriff mit SABLE (vereinfacht nach BLIS Project, 2003)

4.6.3 Building Information Modeling – BIM

Unabhängig von der Art des verwendeten Produktdatenmodells wird in der Branche das Schlagwort BIM – *Building Information Modeling* – verwendet. Die Gebäudedatenmodellierung mittels BIM bedeutet eine bauteilorientierte Erarbeitung eines virtuellen Gebäudemodells. Zusätzlich zu einem dreidimensionalen Geometriemodell sind auch semantische Informationen enthalten. BIM wird auch als Virtual Building Model, digitales Gebäudemodell oder bauteilorientiertes Arbeiten bezeichnet (Mossack, 2005).

Bazjanac (2005) definiert das *Building Information Model* (BIM) als ein instanziiertes Produktdatenmodell, das gewerkübergreifende Informationen zu einem bestimmten Gebäude enthält. Es ist eine statische Darstellung des Gebäudes zu einem bestimmten Zeitpunkt.

Auf Grund der großen Datenmengen kann und soll ein BIM nicht alle das Gebäude betreffende Daten enthalten. Ziel ist es, alle wichtigen und für weitere Bearbeitungen im Gebäudelebenszyklus wiederverwendbaren Daten im BIM zugänglich zu machen. Spezifische Daten, die nur eine einzige Disziplin betreffen, werden in externen Datenbanken gespeichert und können im BIM referenziert werden, z. B. statische Berechnungen, Ergebnisse einer thermischen Simulation.

Die Erstellung eines BIM bedeutet einen beträchtlichen Zeit- und Ressourcenaufwand zu Beginn der Planungsphase. Experten aller Bereiche der Baubranche müssen das Modell gemeinsam erstellen, da ein BIM nur dann von Vorteil ist, wenn die enthaltenen Daten in späteren spezialisierten Softwareanwendungen der einzelnen Branchen wiederverwendet werden können.

Ein BIM kann auf verschiedenen Datenmodellen basieren. Virtuelle Gebäude werden in einigen Softwarepaketen bereits erstellt und in nachfolgenden Anwendungen des

gleichen Herstellers weiterverwendet (Müns und Amend, 2005). Um ein BIM möglichst vielen Anwendern in der Planungs-, Bau- und Nutzungsphase des Gebäudes zugänglich zu machen, ist der Einsatz eines herstellerunabhängigen Produktdatenmodells sinnvoll. Neutrale Schnittstellen ermöglichen herstellerübergreifende Kompatibilität. Als neutrales, von der IAI getestetes Produktdatenmodell können die IFC das Austauschformat eines BIM sein, wie DXF ein Austauschformat für CAD – Zeichnungen ist (Juli, 2005).

Das derzeit einzige, vollständig IFC – kompatible Anwendungsprogramm zur Erstellung eines BIM ist ArchiCAD 9 von Graphisoft (Bazjanac, 2005). Einige andere Anwendungen sind indirekt über eine externe Schnittstelle oder über einen Datenaustausch zu ArchiCAD IFC – kompatibel.

Der Erstellung eines BIM geht die Definition der grundlegenden Datensätze voraus, die im BIM enthalten sein müssen. Daten, die downstream (d. h. von nachfolgenden Anwendungen) benötigt werden und von upstream generierten Daten abhängen, müssen bereits berücksichtigt werden. Ein Beispiel aus der thermischen Simulation wird in Abschnitt 5.4.2 besprochen. Weiters kann das BIM nur jene Daten enthalten, die das IFC – Produktdatenmodell unterstützt. Bei der Erstellung des BIM sind daher Projektteams mit Experten der wichtigsten Gewerke erforderlich.

Vor dem Einsatz muss das BIM in einem sogenannten *Model Checker* überprüft werden (siehe auch Abschnitt 4.7.3). Neben der visuellen Überprüfung der Geometrie können auch projektspezifische Bedingungen, sogenannte *Constraint Sets*, untersucht werden. Diese Sets enthalten u. a. maximale Fluchtwegslängen oder Klassifikationen der verwendeten Baumaterialien (Solibri, 2006).

4.6.4 IFC und VDI

Zwischen den IAI und dem Verein Deutscher Ingenieure (VDI) fand und findet eine aktive Zusammenarbeit statt, um relevante Richtlinien in die IFC zu übernehmen. Eine Richtlinie gilt als „übernommen", wenn der Beschreibungsumfang der VDI Richtlinie abgedeckt wird und die beschriebenen Informationen mit den IFC übertragen werden können (Liebich, 2006). Richtlinien, die von den IFC abgedeckt werden sind:

- VDI 6021: Datenaustausch für die thermische Lastberechnung von Gebäuden
- VDI 6027, Blatt 1: Anforderungen an den Datenaustausch von CAD – Systemen: Gebäude und Gebäudetechnik Konventionen
- VDI 6027, Blatt 2: Anforderungen an den Datenaustausch von CAD Systemen: Anlagentechnik

Derzeit findet eine Zusammenarbeit zwischen IAI und VDI statt, um relevante Teile der VDI 3805 in die IFC zu übernehmen.

4.7 Anwendung

4.7.1 IFC – kompatible Applikationen

Die IFC sind public domain, d. h. sie stehen allen Softwareentwicklern für die Implementierung zur Verfügung. Von der IAI zertifizierte Softwareprodukte sind am Logo der IAI zu erkennen (Abbildung 4.18).

Abbildung 4.18: IAI Logo

Zur Zeit werden IFC2x bzw. IFC2x2 - zertifizierte Softwareanwendungen u. a. von Graphisoft, Autodesk, Bentley, Data Design System, Nemetschek und Granlund angeboten (IAI, 2006c).

4.7.2 Anwenderhandbuch der IAI

Die Anwendung des IFC – Produktdatenmodells erfordert ein Umdenken der Ingenieure und Softwareanwender. Um die Vorteile objektorientierter Gebäudemodelle nutzen zu können, müssen Einsatzmöglichkeiten und Anforderungen an die Erstellung des Modells definiert werden. Das deutsche Chapter der IAI veröffentlichte im April 2006 ein Anwenderhandbuch, das IFC – Import- und Exportfunktionen von CAD – Programmen erläutert (IAI, 2006d). Der Ablauf des Datenaustausches mit folgenden Architektur bzw. Haustechnik – CAD – Programmen wird erläutert:

- Autodesk Architectural Desktop
- Autodesk Revit
- Bentley Architecture
- Graphisoft ArchiCAD
- Nemetschek ALLPLAN
- Data Design System: SHK Partner und Elektropartner
- Hannappel Software GmbH: elcoSystem
- Mensch und Maschine: RoCAD HLSE

Vorgestellt werden auch die Analyseprogramme *IfcStoreyView* und *IfcViewer* des Forschungszentrums Karlsruhe (siehe folgendes Kapitel).

Da die IFC im täglichen Projektgeschäft noch wenig verbreitet sind, werden Einsatzmöglichkeiten des IFC – Modells in verschiedenen Planungsphasen vorgestellt. Mindestanforderungen an den Inhalt des Modells sind im Anwenderhandbuch ebenso definiert wie Aufgaben der Projektbeteiligten.

4.7.3 Analysewerkzeuge

Üblicherweise wird das IFC – Gebäudemodell in einem CAD – Architekturprogramm erstellt. Vor der Weiterverwendung des Modells sollte dieses überprüft werden. Das

Forschungszentrum Karlsruhe stellt dafür verschiedene Werkzeuge frei zur Verfügung.

Mit dem *IfcObjectCounter* können die Elemente innerhalb einer IFC – Datei gezählt werden. Die Syntax und die Semantik des Modells werden überprüft. Über eigene Prüfroutinen kann die Überprüfung des Modells verbessert werden. Vom IfcObjectCounter gefundene Fehler weisen auf Probleme bei der Exportfunktion des Erstellerprogramms hin. Ergebnisse dieser Art der Prüfung sind für den Anwender weniger aussagekräftig, da sie sich auf die verwendeten Systeme beziehen. (Dayal, 2004).

IfcStoreyView ermöglicht die zwei- und dreidimensionale Betrachtung des IFC – Modells. Die Gebäudestruktur, Grundriss sowie Elemente, ihre Eigenschaften und Beziehungen werden dargestellt. Über die *Owner History* kann der Bearbeitungsablauf verfolgt werden. In der IFC – Datei können keine Veränderungen vorgenommen werden. Version 1.7a ist zu IFC2x, IFC2x2, IFC2x3 und ifcXML kompatibel.

Mit dem *IfcViewer* können Modelle schnell visuell überprüft werden. Die Darstellung ist axonometrisch. Das Öffnen eines Modells bedarf einer großen Prozessorleistung.

Ein kommerzielles Überprüfungswerkzeug ist der *Solibri Model Checker* der Firma Solibri. Das Programm ermöglicht sowohl eine Syntaxüberprüfung des Modells als auch die Überprüfung frei wählbarer Parameter. Diese Parameter können projektspezifische Gebäudeeigenschaften sein, z. B. Fluchtweglängen, maximale Raumgröße etc. Dadurch ist es für ein automatisiertes Qualitätsmangagement in der Gebäudeplanung geeignet.

4.7.4 Praxiserfahrungen

Das IFC – Produktdatenmodell befindet sich derzeit noch im Anfangsstadium seiner Entwicklung. Es ist bereits in vielen Anwendungen implementiert, sein Einsatz erfordert jedoch Detailwissen über das Wesen der IFC, derzeitige Möglichkeiten und Einschränkungen. Praktischen Einsatz finden die IFC im Moment in Pilotprojekten mit Expertenunterstützung, um das Modell laufend zu verbessern und auf den breiten Praxiseinsatz vorzubereiten.

Von den IFC – Pilotprojekten werden für diese Arbeit das PM4D – Projekt (Fischer und Kam, 2002) und das disziplinenübergreifende Design – Studio der University of New South Wales (Plume und Mitchell, 2005) herausgegriffen.

4.7.4.1 Pilotprojekt: PM4D

Planung und Bau eines Auditoriums der Helsinki University of Technology wurden im Zuge des PM4D – Projekts (Product Model and Fourth Dimension) abgewickelt. Ziel war es, den Einsatz des IFC – Produktdatenmodells für den Datenaustausch in der frühen Planungsphase sowie der Bauphase zu testen. Neben der dreidimensionalen

Betrachtung des Gebäudes im Gebäudemodell wurde der zeitliche Ablauf in der Bauphase berücksichtigt (4D – Visualisierung).

Die IFC – Schnittstelle 1.5.1 wurde für den Datenaustausch zwischen CAAD – Programmen (ArchiCAD, ALLPLAN), Simulationsprogrammen (RIUSKA, CFX Fluid Dynamics), HVAC – Programmen (MagiCAD) sowie zur 4D – Visualisierung (CPT 4D) und den Datenaustausch zu weiteren Anwendungen verwendet. Der Datenaustausch wurde durch die Weitergabe von IFC – Dateien realisiert.

Fischer und Kam (2002) nennen den reduzierten Zeitbedarf als größten Vorteil des IFC – Einsatzes. Durch den Austausch des geometrischen Modells entfielen erneute Dateneingaben in diesem Bereich. Änderungen am Design konnten einfach weitergegeben werden.

Bazjanac (2002) beschreibt Problemfelder, die in diesem und anderen frühen Pilotprojekten beobachtet wurden:

1. Die Industrie ist nicht auf den Einsatz von Produktdatenmodellen vorbereitet. Vielen Anwendern ist die Nutzung eines dreidimensionalen Gebäudemodells noch fremd. Oft werden aus dem vorhandenen Gebäudemodell nur zweidimensionale Ansichten und Schnitte weiterverwendet. Die Anwendung eines Produktdatenmodells verlangt Expertenwissen, die technische Disziplin und die verwendete professionelle Software betreffend.
2. Die Erstellung des Modells erfordert Wissen über den Ablauf des Datenaustausches und die benötigten Daten der weiteren technischen Planung. Fehlende Daten können eine Folge des eingesetzten *Implementation Views* sein (siehe Abschnitt 4.4.5). Daten fehlen, da sie nicht Teil des verwendeten Views sind. Eine andere Erklärung für fehlende Daten im weitergeleiteten Modell ist in den Aufgaben des Modellerstellers zu suchen. Spezielle Daten sind nicht Teil des Designs sondern werden erst im Laufe der weiteren Projektentwicklung erstellt. Besonders im frühen Designstadium können einige Daten noch nicht ausreichend definiert werden.
3. Anwendungen interpretieren das Gebäudemodell ihren Aufgaben entsprechend verschieden. Um das Gebäudemodell vor dem Einlesen in eine Anwendung aufzubereiten, ist Middleware (Abschnitt 4.4.4) geeignet.
4. Einschränkungen in der Verwendung des IFC – Modells sind sowohl durch das Modell selbst als auch durch kompatible Anwendungen gegeben. Das IFC – Modell ist noch nicht vollständig. Einige Informationen sind noch nicht in der IFC - Spezifikation enthalten, andere nicht in den entsprechenden Anwendungen implementiert. Andererseits können nicht alle Applikationen die Möglichkeiten der IFC voll ausschöpfen. Einige Anwendungen können z. B. auf Grund ihrer Programmierung gekrümmte Wände nicht verarbeiten, obwohl diese in den IFC definiert sind.
5. IFC – Dateien besitzen eine erhebliche Dateigröße. Die Dateigröße erschwert den Datenaustausch und die Bearbeitung der Datei. Die Größe einer IFC – Datei überstieg im PM4D – Projekt die ursprüngliche Größe im anwendungseigenen Format in einigen Fällen um das Fünffache.

6. Vor dem Einsatz der IFC – kompatiblen Software sollte diese überprüft werden. Viele Programmversionen von IFC – Schnittstellen befinden sich noch im Teststadium (Beta Status). Die Anwendungen sollten u. a. auf Funktion, Ausgereifheit, Umfang der Interoperabilität und Herstellersupport überprüft werden. Ein sinnvoller Einsatz der Software hängt weiters vom Können des Anwenders ab. Alle Aufgaben innerhalb eines Projekts die Interoperabilität der Anwendungen erfordern, müssen vor Projektbeginn definiert werden. Die verwendeten Softwareprogramme müssen diese Aufgaben ausführen können.
7. Softwareinteroperabilität funktioniert bereits. Der Projektablauf wird zukünftig durch den Einsatz von Gebäudemodellen entscheidend verändert werden. Die Erstellung des Modells erfordert die Zusammenarbeit aller Disziplinen. Um das Modell in der weiteren Planung sinnvoll einsetzen zu können, müssen Informationsbedürfnisse der nachfolgenden Planer bereits während der Modellerstellung im Architektur – CAD – Programm berücksichtigt werden. Entscheidungen, die normalerweise später im Designprozess getroffen werden, sind vorzuziehen.

4.7.4.2 Pilotprojekt: Multi – disziplinäres Design – Studio

An der University of New South Wales wurde im Zuge einer Projektstudie mit Studenten ein multi – disziplinäres Design – Studio eingerichtet. Die Studenten übernahmen Aufgaben einzelner technischer Disziplinen in der Projektplanung. Der Datenaustausch fand in diesem Fall über einen zentralen IFC – Modell Server statt. Ziel des Projekts war es, allgemeine Probleme bei Verwendung eines Gebäudemodells (BIM) und spezielle Probleme bei der Verwendung eines IFC – basierten Gebäudemodells zu erkennen. Die IFC – Schnittstellen 2.0, 2x und 2x2 wurden verwendet.

Erfahrungen mit dem Gebäudemodell:

- Für den Großteil der verwendeten Programme ist die Definition von Räumen von besonderer Bedeutung. Der Raum stellt die wichtigste Einheit des Gebäudes dar, der Bauteile, Verwendungszweck etc. zugeteilt werden. Verwendet ein Programm spezielle Richtlinien zur Erstellung von Raumnamen oder Raumcodierungen, müssen diese Richtlinien im gesamten Gebäudemodell übernommen werden.
- Die Robustheit des Gebäudemodells muss gewährleistet werden. Geschoßbezeichnungen, korrekte Lagebezeichnungen, der geographische Standort sowie korrekte Bauelemente müssen vorhanden sein und bleiben.
- Der Detaillierungsgrad des Gebäudemodells muss vor Projektbeginn festgelegt werden. Es ist nicht notwendig, dass alle Informationen im Modell gespeichert werden. Anzahl und Genauigkeit der Details im BIM müssen auf die Kosten der Erstellung dieser Details abgestimmt werden.
- Um die semantische Integrität des Modells zu wahren, müssen Objekte der IFC einheitlich genutzt werden. Dies betrifft besonders programminterne Objekte, die in das IFC – Modell übertragen werden, obwohl deren Austausch

nicht beabsichtigt ist. Werden die Objekte von anderen Programmen eingelesen, kommt es zu Missinterpretationen.
- Bauteile sollen mit Materialeigenschaften versehen werden. Spezielle Daten können über den Einsatz von PSETs definiert werden. Wird das Modell in einem Model Checker überprüft, ist darauf zu achten, ob der Model Checker auch PSETs überprüft.

Erfahrungen mit den IFC:

- Obwohl die IFC Softwareinteroperabilität gewährleisten sollen, kann der Einsatz verschiedener IFC – Versionen Inkompatibilität verursachen.
- Der Einsatz von Modell Servern ermöglicht nicht automatisch den Austausch von Teilmodellen. Sollte der Server nur den Up- bzw. Download von gesamten Modellen ermöglichen, ist es sinnvoll das Gebäudemodell in kleinere Modelle (z. B. nach Stockwerken) zu unterteilen, um die Dateigröße gering zu halten.
- Der Einsatz von Modell Servern stellt besondere Aufgaben an das Projektmanagement. Zugriffsprotokolle und die Möglichkeit der Versionierung müssen vorhanden sein. Weiters muss sichergestellt werden, dass alle Beteiligten die vereinbarten Modellstandards des Projekts einhalten. Zugriffs- und Bearbeitungsrechte müssen vergeben werden können, um die Konsistenz des Modells zu wahren. Im Rahmen des Design – Studios standen keine Softwareanwendungen zur Verfügung, die diese Aufgaben übernehmen konnten.

5 IFC in der thermischen Gebäudesimulation

Für die thermische Gebäudesimulation ist die Erfassung der Eingabedaten von besonderer Bedeutung. Die VDI – Richtlinien 6020 und 6021 definieren Anforderungen an Rechenverfahren und Datenaustausch die Gebäudesimulation betreffend. Die Bedeutung des Geometriemodells und Unterschiede in der Betrachtungsweise des Gebäudes durch Architekt und Gebäudetechniker werden ebenso besprochen wie Bedeutung und Einsatzbereiche der IFC in der thermischen Gebäudesimulation.

5.1 Einleitung

Die dynamische Gebäudesimulation wird zur energetischen Optimierung von Gebäuden eingesetzt. Die VDI 6020 definiert die thermisch – energetische Gebäudesimulation als die stundenweise Berechnung der Raumreaktion (Last, Temperatur) unter Berücksichtigung aller inneren und äußeren Einflüsse. Die Modellerstellung, die Bedienung der Simulationsprogramme und die Auswertung der Ergebnisse erfordern Expertenwissen. Um die thermische Gebäudesimulation in der täglichen Planung zu etablieren, müssen diese Bearbeitungsschritte vereinfacht werden.

Jede Gebäudesimulation basiert auf einem Modell, an dem die tatsächlichen Vorgänge nachgebildet werden. Gebäude, Gebäudetechnik und Benutzer sind wichtige Teilbereiche des Modells. Die Genauigkeit der Nachbildung der Wirklichkeit im Modell wirkt sich entscheidend auf die Qualität der Simulationsergebnisse aus. Die Erstellung des Modells bedarf deshalb eines nicht zu unterschätzenden Zeitaufwandes.

5.2 Richtlinien des VDI

5.2.1 VDI 6020: Anforderungen an Rechenverfahren zur Gebäude- und Anlagensimulation

Im Mai 2001 erschien der erste Teil der VDI – Richtlinie 6020, der sich auf die Gebäudesimulation bezieht. Noch nicht erschienen ist Blatt 2, das Anforderungen an die Anlagensimulation definieren wird.

Für die thermische Gebäudesimulation sind verschiedenste Rechenprogramme erhältlich, die sich in ihren Berechnungsmethoden zum Teil stark unterscheiden. Berechnungsalgorithmen und interne Parameter sind für den Anwender kaum zugänglich. Die Simulationsergebnisse sind nicht vergleich- oder prüfbar. Die VDI 6020 legt daher einheitliche Randbedingungen und Mindestanforderungen an die thermische und energetische Simulation fest. Art und Verwendung von meteorologischen Daten, Grundlagen der Bilanz von sensiblen Wärmeströmen im Raum sowie Rechenverfahren für die Ermittlung des instationären Raumverhaltens

werden beschrieben. Die Form der Gebäudebeschreibung (Bauteilphysik und Nutzung) wird ebenfalls einheitlich festgelegt. In diesem Zusammenhang wird der Begriff des thermischen Gebäudemodells verwendet, das im Rahmen der VDI 6021 besprochen wird (Abschnitt 5.2.2).

Die Richtlinie enthält weiters Testbeispiele, die zur Verifikation eines Simulationsprogramms verwendet werden können. Enthalten sind Beispiele zur Überprüfung

- einer Ganzjahressimulation,
- der Raumreaktion auf innere Belastungen bzw. Veränderungen des Sollwerts,
- der Reaktion auf solare Einstrahlung.

Ausgewählte Simulationsprogramme wurden im Rahmen der Richtlinie getestet und werden für den Einsatz in der thermischen Gebäudesimulation empfohlen:

- DOE-2 Version 2.1E
- DS-THERM Version 3.26
- GEBSIMU Version 4.32
- TAS Version 8.0
- TRNSYS Version 14.2, Update 12/98

5.2.2 VDI 6021: Datenaustausch für die thermische Lastberechnung von Gebäuden

Die VDI – Richtlinie 6021, erschienen im März 2000, beinhaltet ein Datenaustausch-verfahren zwischen CAAD – Systemen und Softwareprogrammen der technischen Gebäudeplanung. Der Datenaustausch von Gebäudeinformationen wird aus dem Blickwinkel der thermischen Lastberechnung betrachtet. Dazu wird ein thermisches Gebäudemodell beschrieben, das unabhängig von den verwendeten Rechenverfahren ist. Anwendungsbereiche sind dadurch nicht nur die thermische Gebäudesimulation sondern auch Kühllast- und Wärmebedarfsberechnungen und der Wärmeschutz-nachweis.

Der Datenaustausch umfasst Gebäudeinformationen in Form von geometrischen Daten und Bauteilinformationen. Die Richtlinie beschreibt ein Gebäudemodell, das auf die thermische Betrachtung abgestimmt ist. In diesem Zusammenhang spricht die Richtlinie von einem *„Aspektmodell der Lastberechnung*". Das thermische Gebäudemodell gliedert sich in Gebäude – Ebene – Raum. Einzelne Räume sind über gemeinsame Umschließungsflächen gekoppelt. Neben der Untergliederung der wärmespeichernden Bauteile in Bauteiltypen, werden Bauteile auch nach Art der Randbedingungen unterteilt. Eine Randbedingung bezeichnet z. B. die Raumtemperatur des angrenzenden Raums. Eine durchgehende Wand gleichen Bauteiltyps wird dadurch in zwei Teilbereiche mit verschiedenen angrenzenden Raumtemperaturen geteilt. Das Aspektmodell wird in drei Schritten gebildet:

1. Ein Raum wird durch seine Raumumschließungsflächen definiert. Jede Umschließungsfläche besitzt eine Flächennormale.
2. Teilflächen werden gebildet, die von der Geometrie hinter der Flächennormalen abhängen.
3. Räume mit gemeinsamen Teilflächen werden gekoppelt.

Die Bildung der Teilflächen in Abhängigkeit der angrenzenden Raumtemperaturen ist in Abbildung 5.1 dargestellt. Den so entstehenden Bauteilflächen werden im nächsten Schritt Bauteileigenschaften zugewiesen. Bauphysikalische Eigenschaften können in wärmespeichernde und nicht – wärmespeichernde Eigenschaften unterschieden werden.

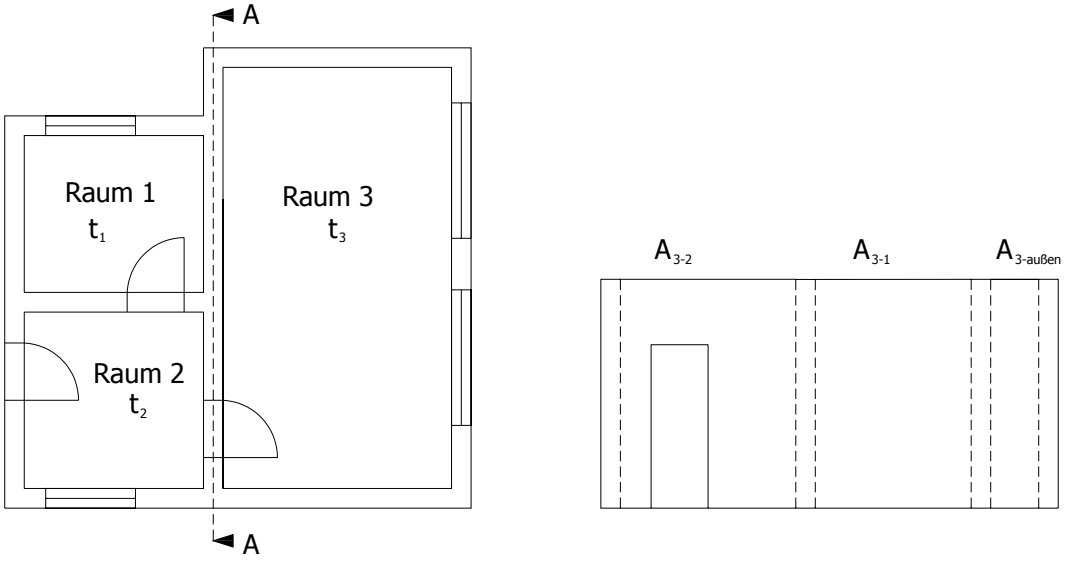


Abbildung 5.1: Erstellung von Teilflächen im Aspektmodell

Nutzungsbedingung können nicht immer aus dem CAD – System gewonnen werden. Die Übertragung der Nutzungsbedingungen ist in dieser Richtlinie auf den Wärmeschutznachweis, den Wärmebedarf nach DIN 4701 und die Kühllastberechnung nach VDI 2078 abgestimmt. Die Beziehungen zwischen Gebäude, Ebene und Raum sowie den Bauteilen und den Nutzungsbedingungen können in einem Strukturmodell übersichtlich dargestellt werden. Abbildung 5.2 zeigt die logische Struktur des thermischen Gebäudemodells und ist der VDI 6021 entnommen.

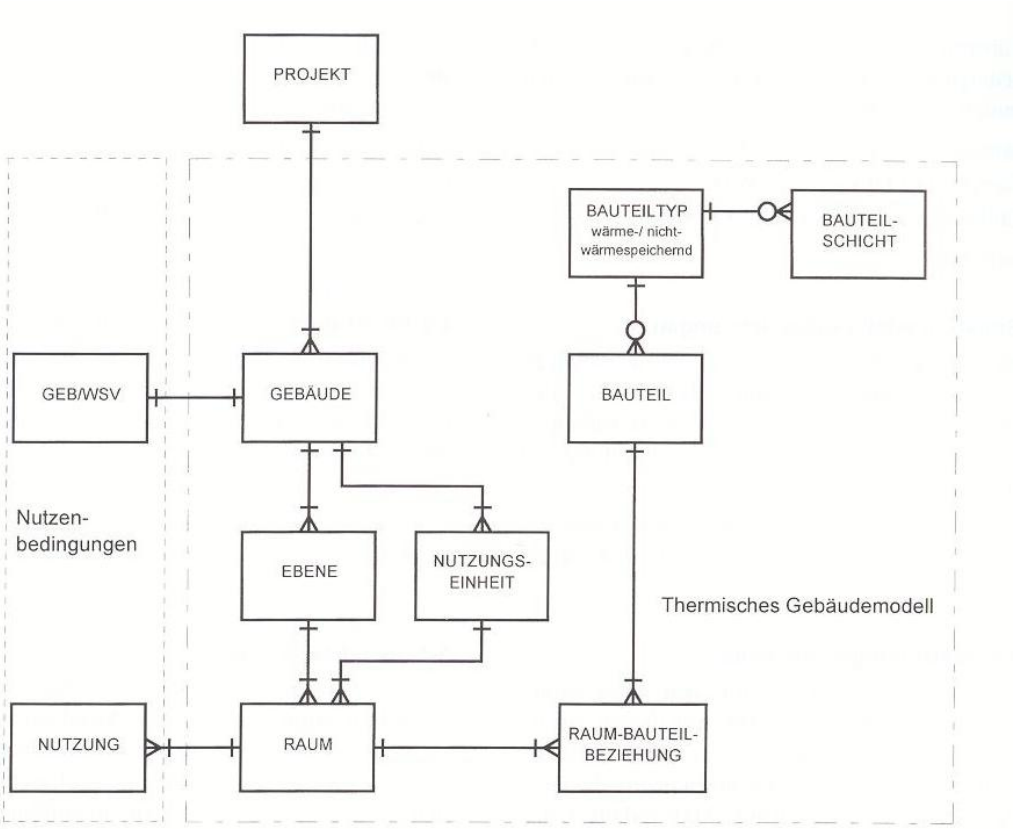
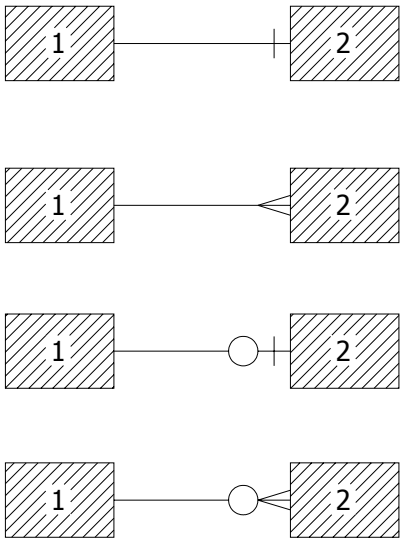


Abbildung 5.2: Strukturmodell des thermischen Gebäudemodells (VDI 6021)

Die Kennzeichnungssystematik zur Beschreibung der Beziehungen zwischen Informationseinheiten ist in Abbildung 5.3 dargestellt. Aus der Kennzeichnung ergibt sich beispielsweise, dass ein Gebäude genau einem Projekt zugewiesen ist. Ein Projekt kann jedoch aus mehreren Gebäuden bestehen.



Die Informationseinheit 1 hat eine Beziehung zur Informationseinheit 2.

Die Informationseinheit 1 hat eine Beziehung zu einer oder mehreren Informationseinheiten 2.

Die Informationseinheit 1 hat eine Beziehung zu keiner oder einer Informationseinheit 2.

Die Informationseinheit 1 hat eine Beziehung zu keiner, einer oder mehreren Informations-einheiten 2.

Abbildung 5.3: Kennzeichnung des Strukturmodells nach VDI 6021

Für den Austausch des thermischen Gebäudemodells definiert die VDI 6021 ein Datenaustauschformat. Die Spezifikation ist in Form eines EXPRESS bzw. EXPRESS –

G Schemas offengelegt. Die Schnittstelle basiert daher auf dem STEP – Standard. Der Datenaustausch findet über SPF – Dateien statt. Bei der Entwicklung der Richtlinie fand eine Zusammenarbeit mit der IAI statt. Der Beschreibungsumfang der VDI 6021 wird von den IFC abgedeckt. Es können dieselben Informationen übertragen werden (Liebich, 2006).

5.3 Das geometrische Gebäudemodell der Simulation

Die Erfassung des Gebäudemodells nimmt einen großen Teil des Arbeitsaufwandes einer Gebäudesimulation ein. Redundante Dateneingaben erhöhen den Zeitaufwand und sind eine mögliche Fehlerquelle. Neben den klassischen Varianten der Nachbildung besteht auch die Möglichkeit des Imports von Teilmodellen oder des gesamten Gebäudekomplexes. Hierfür stehen direkte Schnittstellen, graphische Benutzeroberflächen und Produktdatenmodelle, die Softwareinteroperabilität ermöglichen, zur Verfügung.

5.3.1 Bedeutung des Geometriemodells

Das Modell eines Gebäudes bildet den Kern jeder thermischen oder energetischen Simulation. Der Übereinstimmung des Modells mit der Wirklichkeit und eventuell getroffenen Vereinfachungen kommen daher große Bedeutung zu. Zusätzlich muss auf Grund der Berechnungsmethoden einer Simulation das Modell dreidimensional vorliegen. Dies ist bei vorhandenen Zeichnungen größtenteils nicht der Fall. In den meisten Fällen werden zur Geometrieerfassung im Simulationsprogramm Geometriedaten aus Architekturplänen neu erfasst und manuell eingegeben. Das bedeutet, dass bereits existierende Daten kopiert werden müssen. Dieser Vorgang stellt eine potentielle Fehlerquelle dar. Zudem ist bei größeren und komplexeren Gebäuden der Aufwand der Akquisition der Geometrie oft ein Hauptgrund für den Abbruch einer Simulation.

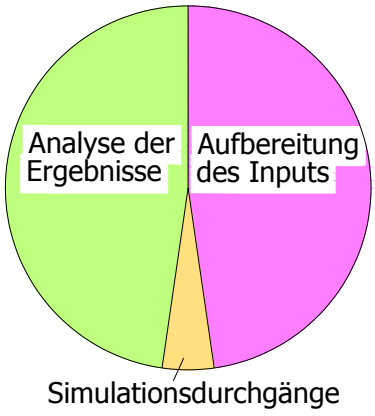


Abbildung 5.4: Verteilung der Gesamtkosten einer Energiesimulation (nach Bazjanac, 2001)

Betrachtet man die Gesamtkosten einer thermischen oder energetischen Simulation, teilen sich diese gleichmäßig auf die Aufbereitung der Eingabedaten und die Auswertung der Ergebnisse auf. Die Kosten der eigentlichen Simulationen (Berechnungen mit dem Computer) sind minimal (siehe Abbildung 5.4).
Die Vorbereitungen bis zum ersten, erfolgreichen Simulationsdurchgang lassen sich in mehrere Teilschritte zerlegen. Erst nach dem erfolgreichen Testlauf können Parameterstudien oder aussagekräftige Varianten durchgerechnet werden.

Ein Großteil der Vorbereitungen entfällt auf die Aufbereitung der Eingabedaten (siehe Abbildung 5.5). Fehlerbehebung und Analyse der Ergebnisse sind mehrmals notwendig bis ein erstes, verwendbares Simulationsergebnis vorliegt.

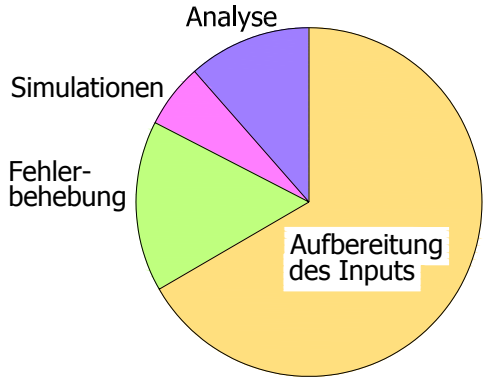


Abbildung 5.5: Verteilung des Aufwands bis zum ersten erfolgreichen Simulationsdurchgang (nach Bazjanac, 2001)

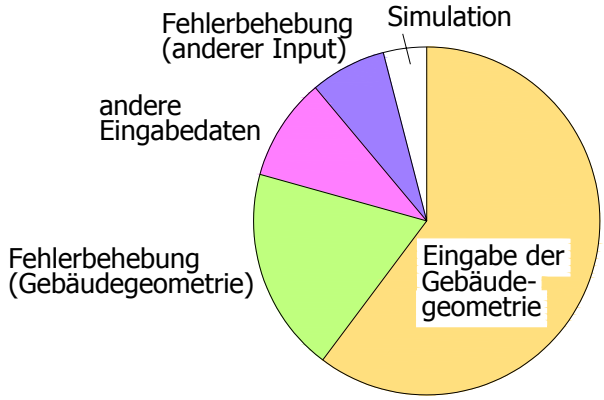


Abbildung 5.6: Anteil der Gebäudegeometrie an der Aufbereitung der Eingabedaten (nach Bazjanac, 2001)

Bei der Analyse der Teilschritte, die zur Aufbereitung aller Eingabedaten notwendig sind, zeigt sich, dass die Geometrieerfassung den größten Anteil einnimmt (siehe Abbildung 5.6). Neben der Ersterfassung der Geometrie bedeutet auch die Fehlerbehebung im geometrischen Modell einen erheblichen Aufwand. Der tatsächliche Aufwand der Nachbearbeitung hängt stark von der Größe und Komplexität des Projekts ab. Die Ziele der durchgeführten Simulation, Ausbildung und Erfahrung des Nutzers sowie Zeit- und Geldbudgets des Projekts spielen ebenfalls eine wichtige Rolle.

Bazjanac (2001) misst den Auswirkungen der aufwändigen Geometrieerfassung große Bedeutung zu. Die langwierige und fehleranfällige Akquisition kann dazu führen, dass der eigentliche Sinn einer thermisch – energetischen Simulation nicht weiter verfolgt wird. Die Untersuchung von Varianten wird eingeschränkt und nach einigen Simulationsdurchgängen werden Ergebnisse oft als definitive Antworten interpretiert.

Diese Bedeutung der Geometrie führt dazu, dass Vereinfachungen in diesem Bereich positive Auswirkungen auf den gesamten Simulationsprozess haben, z. B. durch den teilweisen Import von Geometrien. Die Gebäudesimulation wird leichter zugänglich und kann von breiteren Anwenderschichten genutzt werden.

In den folgenden Abschnitten wird auf die Möglichkeiten der Geometrieerfassung eingegangen. Neben Softwareinteroperabilität und genormten Austauschformaten, werden auch andere Methoden zur Geometrieerfassung vorgestellt.

5.3.2 Nachbildung der Gebäudegeometrie

Die gängigste Form der Geometrieerfassung stellt die Nachbildung dar. Aus zweidimensionalen Zeichnungen wird im Simulationsprogramm ein neues, dreidimensionales Modell erstellt. Bereits auf Datenträger vorhandene Daten werden nicht automatisch übernommen, sondern manuell übertragen.

Voraussetzung der Nachbildung ist die Kenntnis der Syntax des Simulations-programms. Diese Methode der Nachbildung ist weit verbreitet, wenn auch fehleranfällig und arbeitsintensiv.

Falls vorhanden kann eine graphische Benutzeroberfläche (Graphic User Interface, GUI) zur Nachbildung der Geometrie verwendet werden. Vorhandene Daten werden über das GUI manuell eingetragen. Auch bei dieser Methode werden zweidimensionale Daten durch die manuelle Eingabe in ein dreidimensionales Modell überführt. Die maximale Komplexität des Modells hängt von den Möglichkeiten des GUI ab. Werden Schablonen verwendet, ist die Form des Gebäudes auf diese beschränkt. Durch den Einsatz eines GUI entfällt die manuelle Eingabe der Programmsyntax.

5.3.3 Import bestehender Gebäudegeometrien

Aus Sicht des Benutzers einer Simulationssoftware erleichtert schon der teilweise Import von bestehenden Geometriedaten die Ausführung. Importe können über direkte Schnittstellen oder graphische Benutzeroberflächen realisiert werden. Die umfassendste Methode bilden Produktdatenmodelle, die Softwareinteroperabilität und dadurch direkten Datenaustausch ermöglichen.

5.3.3.1 Direkte Schnittstelle

Über eine direkte Schnittstelle zwischen dem Simulationsprogramm und der Datenquelle kann die Gebäudegeometrie „nahtlos" übertragen werden. Als Quelle können CAD – Applikationen oder Datenbanken fungieren. Die Schnittstelle muss die Datenstrukturen von Zielprogramm und Quelle verstehen. Die Hauptaufgabe des Interface liegt in der „Übersetzung" der Geometrie in die Syntax der Simulationsapplikation.

Aus dieser Übersetzungsaufgabe folgt, dass Schnittstellen für den Geometrieimport auf „Knopfdruck" hoch spezialisiert sind und nur bei kompatiblen Applikationen anwendbar sind. Schnittstellen dieser Art sind wenig verbreitet und teuer.

5.3.3.2 Graphische Benutzeroberfläche

Graphische Benutzeroberflächen (GUI) oder Präprozessoren, die dreidimensionale Geometrien lesen und in Eingabedaten des Simulationsprogrammes übersetzen können, ermöglichen ebenfalls den Import bestehender Daten. Zur Zeit erhältliche GUIs importieren keine kompletten 3D – Gebäude oder benötigen manuelle Korrekturen.

Ein Hauptgrund in der erforderlichen Nachbearbeitung des Imports liegt in der grundlegend verschiedenen Betrachtungsweise des Gebäudes durch Architekt und Fachplaner. Das Gebäudemodell für die thermische Simulation beinhaltet weniger Informationen als das gesamte Modell. Andere Bereiche müssen hingegen genauer dargestellt werden (z. B. die Zuteilung von Wänden zu verschiedenen thermischen Zonen).

Um ein vorliegendes Gebäudemodell in die Simulation zu integrieren, ist in den meisten Fällen eine manuelle Nachbearbeitung notwendig, um das Modell sowohl zu vereinfachen als auch in manchen Bereichen zu detaillieren. Beispiele für Eingriffe in das übernommene Modell sind:

- Die Annäherung gekrümmter Flächen durch flache Bauteile, da die meisten Simulationsprogramme gekrümmte Flächen nicht kennen.
- Die Gruppierung sich wiederholender Elemente oder Beschreibungen von Räumen und Oberflächen.
- Die Auswahl relevanter Teile des Gebäudes und das Entfernen überflüssiger Informationen (z. B. Innenwände zwischen Räumen gleicher thermischen Zonen).
- Das Hinzufügen von fehlenden geometrischen Informationen (z. B. die dreidimensionale Lage von Sensoren).
- Die thermische Zonierung von Räumen gleicher Nutzung.

Ein für den Geometrieimport geeignetes und nützliches GUI sollte nach Bazjanac (2001) folgende Eigenschaften besitzen:

- Der Import jeder dreidimensionalen Geometrie muss möglich sein. Schablonen würden die Komplexität der Gebäude einschränken.
- CAD – Daten sollten in ihrem ursprünglichen Format eingelesen werden, um Fehler bei der Auslesung zu vermeiden.
- Die Änderung der Eingabedaten soll in einem ASCII – Texteditor möglich sein. Das Einfügen von bereits bestehenden Segmenten anderer Simulationen wird ermöglicht.
- Schnittstellen zu externen Datenbanken und Materialbibliotheken sollen vorhanden sein.

CAD – Anwendungsprogramme können für den Einsatz als GUI adaptiert werden. Sehr gute Programmkenntnisse sind für den Nutzer Voraussetzung.

5.3.3.3 Softwareinteroperabilität

Der Einsatz eines herstellerunabhängigen Datenmodells ist der Ansatz der Softwareinteroperabilität. Der direkte Datenaustausch zwischen den Anwendungen wird ermöglicht.

Das Modell der IFC unterstützt dreidimensionale Geometrien. Gebäudemodelle, die am Anfang der Designkette erstellt wurden, können in Programme, die auf dieser Geometrie aufbauen, importiert werden. Durch den Einsatz von Vermittlungssoftware werden nur jene Informationen importiert, die für die Simulation relevant sind.

Ein Import auf „Knopfdruck" ist auch bei Anwendung der IFC unrealistisch, da das architektonische Modell in Abhängigkeit der speziellen Anforderungen eines Projekts auf das Simulationsprogramm abgestimmt werden muss. Redundante Dateneingaben, Fehlerbehebungen im Geometriemodell und die Nachbildung des Modells in der Programmsyntax entfallen allerdings und machen die thermische und energetische Simulation einfacher und schneller in der Handhabung.

Der Einsatz des IFC – Modells in der thermischen Gebäudesimulation wird im folgenden Kapitel betrachtet.

5.4 Bedeutung der IFC für die thermische Simulation

5.4.1 Einsatzbereiche

Die Analyse eines Gebäudes mit Hilfe einer thermischen Simulation erfordert eine große Menge an Daten, die in ein Simulationsprogramm eingegeben werden müssen. Zu diesen Angaben zählen neben der Gebäudegeometrie auch Baumaterialien, haustechnische Anlagen und ihre Betriebszeiten sowie innere Wärmelasten und ihr zeitlicher Verlauf.

Traditionell werden die Eingabedaten einer Gebäudesimulation aus einer Vielzahl von Quellen zusammengetragen und händisch in ein unabhängiges Simulationsprogramm eingegeben. Obwohl die Eingabeoberflächen der Programme laufend verbessert und benutzerfreundlicher wurden, änderte sich am Prinzip der aufwändigen Datenerfassung nichts (Hitchcock, 2002). Weiters stehen Daten eines unabhängigen Simulationsprogramms keinem anderen Gebäudeanalyseprogramm zur Verfügung. Auch ähnliche Anwendungen (z. B. Tageslichtsimulationen) erfordern eine neuerliche, manuelle Dateneingabe.

Das Hauptaugenmerk bei der Implementierung des Datenmodells der Industry Foundation Classes in Gebäudesimulationsprogramme liegt in der Übertragung der Gebäudegeometrie. Dies ist keine Folge von unzureichenden Definitionen anderer Daten, wie Materialien in den IFC. Die Geometrie ist jedoch der kleinste, gemeinsame Nenner aller IFC – kompatiblen Anwendungen. Sie ist in jeder instanziierten IFC – Datei enthalten. Weiters bedeutet die Erstellung der Gebäudegeometrie bei der Dateneingabe in Simulationsprogramme den größten Zeit- und Ressourcenaufwand.

Eine Übertragung bestehender Geometrie aus CAD – Anwendungen lohnt sich daher besonders.

Besonders umfangreiche Gebäudemodelle setzen zudem eine umfangreiche IFC – Schnittstelle der Applikationen voraus, wenn das Modell vollständig genutzt werden soll. In manchen Fällen ist es einfacher, sehr detaillierte Informationen in der betreffenden Applikation manuell einzugeben. Als Beispiel seien hier Beschattungseinrichtungen von Fenstern genannt, die im Simulationsprogramm aus einer internen Datenbank hinzugefügt werden können (Jokela in Fischer und Kam, 2002).

Nichtsdestotrotz ermöglicht die HVAC Domain der IFC die Übertragung detaillierter Anlagendaten. Dieses Erweiterungsschema wurde an die Anforderungen der Gebäude- und Anlagensimulation angepasst. Unterstützt werden u. a. die dynamische Lastberechnung, das Design von Anlagen sowie die Komponentenauswahl und der Produktdatenaustausch von Komponenten (Bazjanac, 2003). Die Übertragung von Regelungseinrichtungen befindet sich noch in einem Anfangsstadium. Voraussetzung für die Nutzung von im IFC – Modell beschriebenen haustechnischen Anlagen ist, dass sowohl in der sendenden als auch in der empfangenden Applikation das HVAC Erweiterungsschema implementiert ist.

Bazjanac und Crawley (1997) sehen vier grundlegende Vorteile des IFC – Einsatzes für die thermisch – energetische Gebäudesimulation:

- Nahezu kostenloser Zugriff auf gebäude- und bauteilbezogene Daten
- Stark verkürzte Simulationszeiten
- Einfachere Einbindung anderer technischer Disziplinen in die Simulation und Analyse des Gebäudes
- Verbesserte Nutzungsmöglichkeiten der Simulationsergebnisse

5.4.2 Das Gebäude aus Sicht der thermischen Simulation

Da der vollständige Datenbestand eines Gebäudemodells weit größer ist als von einer einzigen Anwendung benötigt, wird das Modell in sogenannte Sichten oder *Views* unterteilt. In den IFC sind verschiedene Views definiert. Neben der architektonischen Sicht des Gebäudes, wird auch eine thermische Sicht verwendet. Diese thermische Sicht beschreibt alle Daten, die den passiven, thermischen Eigenschaften des Gebäudes zugeordnet werden und für thermische Simulationen wesentlich sind. Ein Simulationsprogramm übernimmt die thermische Sicht für Berechnungen und ignoriert andere Betrachtungen des Gebäudes.

Das Gebäude unterscheidet sich in architektonischer Hinsicht stark von der thermischen (vergleiche auch VDI 6021, Abschnitt 5.2.2). Einzelne, voneinander getrennte Räume werden zu thermischen Zonen zusammengefasst. Um die Wärmeleitung durch Wände richtig berechnen zu können, müssen Raumbegrenzungen den entsprechenden Zonen zugewiesen werden. Eine aus architektonischer Sicht durchgehende Wand grenzt an verschiedene thermische Zonen (Abbildung 5.7). Die durchgehende Konstruktion muss für die thermische Simulation segmentiert und den entsprechenden Zonen zugewiesen werden. Eine

weitere Segmentierung ist erforderlich, wenn ein Wandteil auf einer Seite einer einzigen Zone, auf der anderen Seite zwei verschiedenen Zonen angehört. Im Beispiel ist dies bei den Zonen 1, 2 und 4 dargestellt. Die Begrenzung von Zone 1 muss auf Seiten der Zonen 2 und 4 weiters unterteilt werden.

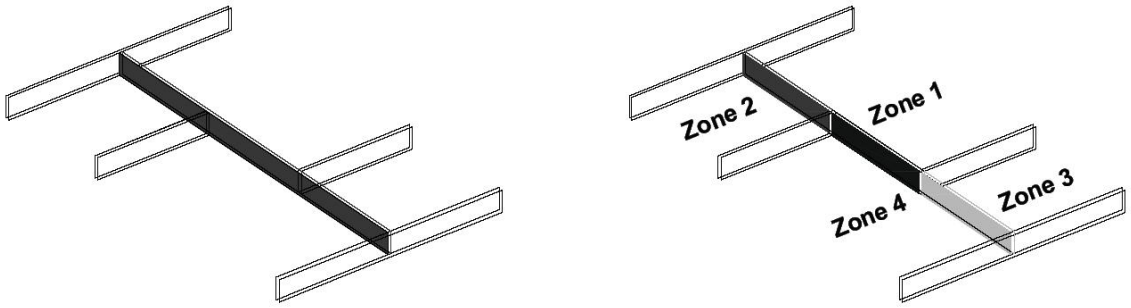


Abbildung 5.7: Definition der Raumbegrenzungen im BIM (Bazjanac, 2005)

Diese thermische Zonierung erfordert einen großen Arbeitsaufwand, wenn sie manuell im Simulationsprogramm durchgeführt werden muss. Für komplexere Gebäude sind daher nur jene Modelle hilfreich, die diese Zonierung und Aufteilung der Raumbegrenzungen automatisch durchführen (Bazjanac, 2005).

Simulationsprogramme, die Wärmeflüsse durch Wände nicht berücksichtigen oder von gleichen Temperaturniveaus der Zonen ausgehen, erfordern eine weniger detaillierte Segmentierung der Wände. Anwendungen auf dem Stand der Technik setzen jedoch diese Art der Zonierung voraus. Erwähnt sei hier das Simulations-programm EnergyPlus, das zur Zeit einzige dieser Art, das direkt IFC – kompatibel ist. EnergyPlus wird in Kapitel 6.3.6 genauer beschrieben.

Obwohl das Gebäudemodell der thermischen Sicht detaillierter ist, was Zonierung und Raumabgrenzungen betrifft, kann das geometrische Modell für die Simulation stark vereinfacht werden. Gebäudesimulationsprogramme betrachten ebene Flächen in ihrer Länge, Breite, Tiefe und Position. Zusätzliche im BIM definierte Geometrien sind für Simulationen dieser Art nicht relevant. Für den Import der Gebäude-geometrie eignen sich daher Vermittlungsprogramme, sogenannte *Middleware Tools*, zur Vereinfachung der Geometrie vor dem Import aus dem IFC – Modell. Das Middleware Tool BSPro COM von Olof Granlund Oy wird in Abschnitt 4.4.4 genauer betrachtet.

5.4.3 IFC – Kompatibilität

Um ein IFC – basiertes Gebäudemodell (BIM) in der thermischen Simulation verwenden zu können, muss das Simulationsprogramm IFC – kompatibel sein. Die Kompatibilität wird durch die verwendete IFC – Version des Gebäudemodells beeinflusst. Simulationsprogramme, die ältere Versionen verwenden sind als nicht direkt kompatibel anzusehen. Allgemein werden drei Gruppen von Anwendungen unterschieden: direkte, indirekte und nicht kompatible Applikationen.

Direkt kompatible Programme können relevante Daten aus dem Modell in das interne Datenformat der Simulationssoftware übertragen. Sie besitzen eine eigene IFC – Schnittstelle derselben Version wie das BIM und ermöglichen Daten aus einer IFC – Datei auszulesen bzw. Daten in das BIM zu schreiben. Programme ohne IFC – Schnittstelle besitzen keine Interoperabilität mit dem Gebäudemodell. Es kann zwischen indirekt kompatiblen und nicht kompatiblen Programmen unterschieden werden.

Indirekt kompatible Anwendungen besitzen ein IFC – Interface, das auf einer veralteten Version beruht (z. B. IFC2x oder IFC2.0). Sie können entsprechend der verwendeten IFC – Version nur teilweise Daten auslesen oder übertragen. Werden sie in Verbindung mit einer direkt kompatiblen Applikation verwendet, kann die Datei in der aktuellen Version gespeichert und in das BIM übertragen werden.

Nicht kompatible Anwendungen besitzen keine IFC – Schnittstelle. Sie können weder Daten aus dem BIM auslesen noch in das BIM importieren. Diese Art der Software kann indirekte Interoperabilität mit dem Gebäudemodell erreichen, wenn Ergebnisse in ein direkt kompatibles Programm exportiert werden können. Für diesen Export kann ein anderes Format als die IFC verwendet werden. Über die direkt kompatible Software können Ergebnisse in das Gebäudemodell übertragen werden. Abbildung 5.8 zeigt die angesprochenen Möglichkeiten der Kompatibilität von Softwareanwendungen zum IFC – Gebäudemodell. Die Wege der Datenübertragung basieren auf dem Austausch physischer Dateien. Befindet sich das BIM auf einem Modell Server spielen die verwendeten IFC – Versionen eine untergeordnete Rolle. Indirekt kompatible Software kann in diesem Fall zuweilen auch ohne Vermittlung durch ein direkt kompatibles Programm auf das Gebäudemodell zugreifen.

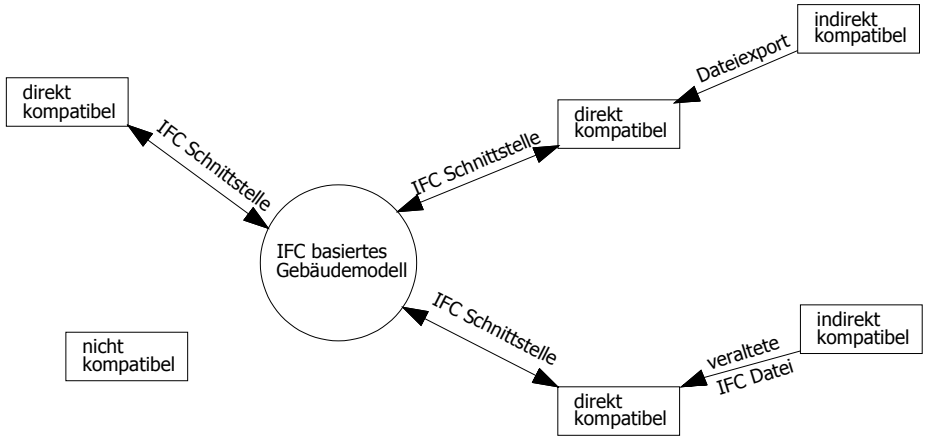


Abbildung 5.8: Schnittstellen und Datentransfermöglichkeiten zu einem BIM (nach Bazjanac, 2005)

Ein Beispiel für den dargestellten Pfad eines direkt kompatiblen Programms ist RIUSKA. Das Programm liest IFC – Dateien direkt ein. Ein mittels Dateiimport indirekt kompatibles Programm ist AutoCAD. Eine AutoCAD – Zeichnung kann als dwg – Datei exportiert werden. Das Haustechnikprogramm DDS liest die dwg – Datei ein und erstellt ein IFC – Gebäudemodell. DDS ist direkt IFC – kompatibel und ermöglicht über seine dwg – Schnittstelle indirekte Kompatibilität für AutoCAD.

5.4.4 IFC – kompatible Simulationsprogramme

IFC – Modelle bieten den großen Vorteil der einfachen Geometrieaquirierung über ein herstellerunabhängiges Datenformat. IFC – Schnittstellen für Gebäudesimulationsprogramme werden laufend entwickelt. Hier sollen einige IFC – kompatible Simulationsprogramme vorgestellt werden, die international sowohl für Forschung als auch Planung von Bedeutung sind.

EnergyPlus ist eines der weitest entwickelten Simulationsprogramme und das einzige seiner Art das direkt kompatibel zu IFC2x2 ist. Es importiert IFC – Gebäudemodelle (BIM) über den BS Pro COM – Server. Speziell für EnergyPlus wurde die Client Software *IFCtoIDF* entwickelt, um die Gebäudegeometrie den Bedürfnissen von EnergyPlus anzupassen und Eingabedateien im programmeigenen Format zu generieren. Neue Daten können über diese Tools hingegen nicht in ein BIM übertragen werden. Über ein IFC HVAC – Interface, das ebenfalls vom LBNL entwickelt wurde, können HVAC relevante Definitionen und Profile aus dem BIM importiert bzw. in das BIM exportiert werden. Simulationsergebnisse direkt von EnergyPlus in das bestehende BIM zu übertragen, ist nicht vorgesehen. Mit Hilfe eines weiteren Programms (SMIET) können externe Dateien, die das gesamte Datenvolumen der Simulation enthalten, im BIM referenziert werden. SMIET ist direkt kompatibel zu IFC2x2 und wird für die Datenübertragung von jenen Daten benutzt, die nicht über die Interfaces von EnergyPlus übertragen werden können (Bazjanac, 2005).

Besonders in Nordamerika ist DOE-2 weit verbreitet. Die Eingabedaten von DOE-2 können teilweise in das Format von EnergyPlus übertragen werden. Dadurch wird DOE-2 indirekt IFC – kompatibel. RIUSKA ist eine graphische Benutzeroberfläche von DOE-2. Über die Vermittlungssoftware BS Pro COM – Server werden die Gebäudegeometrie und Raumdefinitionen übertragen. RIUSKA ist direkt kompatibel zu IFC2x2. Die Simulationsmöglichkeiten von RIUSKA sind gegenüber EnergyPlus weit geringer (vergleiche die Abschnitte 6.3.6 und 6.6).

Das Gebäude- und Anlagensimulationsprogramm TRNSYS ist indirekt IFC – kompatibel. Mit Hilfe des CAD – Programms SimCAD können TRNSYS – Eingabedaten der Gebäudegeometrie generiert werden. SimCAD ermöglicht den Import von DXF – und IFC – Dateien (TRNSYS, 2006).

Seit 1994 entwickelt das Fraunhofer – Institut FIRST in Zusammenarbeit mit der Technischen Universität Berlin das Simulationsprogramm SMILE. Das Programm wurde für die energetische Gebäude- und Anlagensimulation entwickelt. Luftzirkulation und Wärmeverteilung können simuliert werden. Es besteht die Möglichkeit, Gebäudemodelle über eine IFC – Schnittstelle zu importieren (FIRST, 2006).

6 Gebäudesimulation mit RIUSKA

Dieses Kapitel beschäftigt sich mit dem praktischen Einsatz der IFC in der Gebäudesimulation. Nach einer Übersicht des Simulationstools RIUSKA und einer detaillierten Beschreibung des Simulationskerns DOE-2, wird die verwendete IFC – Schnittstelle betrachtet.

6.1 Einleitung

RIUSKA ist eine Softwareanwendung zur dynamischen Simulation von Behaglichkeit und Energieverbräuchen von Gebäuden. 1996 begann die finnische Planungs- und Beratungsfirma *Olof Granlund Oy* mit der Entwicklung, um den Einsatz der thermischen Gebäudesimulation im täglichen Projektgeschäft zu vereinfachen.

Das Programm richtet sich an Planungsingenieure und setzt keine theoretischen Kenntnisse der thermischen Gebäudesimulation voraus. RIUSKA stellt eine benutzerfreundliche Windows – Oberfläche für das bestehende Simulationsprogramm DOE-2 dar. Die aufwendige Erstellung von ASCII – Eingabedateien für das Simulationsprogramm übernimmt RIUSKA ebenso wie die Auswertung der Ergebnisse. Der Planungsingenieur kann sich auf die Interpretation der Ergebnisse konzentrieren.

Hauptbestandteile von RIUSKA sind neben dem Simulationskern DOE-2, die Benutzeroberfläche, eine Simulationsdatenbank sowie Module für Geometrieerfassung und Auswertung. Der Simulationskern wird ausführlich in Kapitel 6.3 besprochen. Die Simulationsdatenbank stellt einen wichtigen Teil des Programms dar. Sie enthält alle für die thermische Gebäudesimulation und Energiekostenrechnung notwendigen Daten, sowie Ergebnisse bereits durchgeführter Simulationen. Die graphische Benutzeroberlfläche ermöglicht die einfache Übersicht des Projekts, stellt die Geometrie dar und ist Portal für den Zugriff auf Material-, Wetter- und Lastdatenbanken. Ebenso werden in diesem Interface für fehlende Daten Default – Werte generiert. Ergebnisse können direkt in RIUSKA abgefragt oder als druckfertige Berichte in Excel ausgegeben werden. Textdateien mit stündlichen Ergebnissen können ebenfalls erstellt werden.

Die Erfassung der Geometrie erfolgt durch den Import eines vollständigen Gebäudemodells. RIUSKA übernimmt die Gebäudedaten über eine IFC – Schnittstelle. Unabhängig davon hat Granlund das AutoCAD basierte Modul SMOG (Space Modeller by Olof Granlund Oy) entwickelt. Es ermöglicht eine einfache Erstellung der Geometrie. In dieser Arbeit wird statt SMOG der Geometrieimport über die IFC – Schnittstelle verwendet.

6.2 Anwendungsbereiche von RIUSKA

Die Entwickler sehen vier Einsatzbereiche für RIUSKA (Jokela, 1997). Zu Beginn der Planungsphase (1) werden grundlegende Entscheidungen über die Gestaltung des Gebäudes getroffen. Eine thermische Jahressimulation kann hier Auswirkungen verschiedener Designs auf den erwarteten Energieverbrauch verdeutlichen. Im weiteren Verlauf der Gebäudeplanung (2) unterstützt RIUSKA die Anlagendimensionierung. Nach Auswahl des endgültigen Gebäudedesigns und der Dimensionierung der haustechnischen Anlagen (3) können mit RIUSKA genauere Aussagen über den Gesamtenergieverbrauch getroffen werden. Diese Werte sind genauer als erste Berechnungen während der frühen Planungsphase. Im laufenden Betrieb (4) dient das Programm zur Unterstützung des Facility Managements. Die Simulation der Auswirkungen von Sanierungsmaßnahmen sei hier erwähnt.

RIUSKA bietet zwei Möglichkeiten einer thermischen Gebäudesimulation:

- Behaglichkeitssimulation
- Energiesimulation

Bei einer *Behaglichkeitssimulation* wird die Raumlufttemperatur auf Stundenbasis raumweise ermittelt. Einsatzbereiche der Behaglichkeitssimulation sind:

- Vergleich der Raumluftqualität bei Einsatz verschiedener Systeme
- Parameterstudien über Fenstertypen, Beschattung
- Problemanalyse in bestehenden Gebäuden.

Die Anlagensimulation wird in RIUSKA als *Energiesimulation* bezeichnet. Die stündlichen Berechnungsergebnisse können als Grundlage der Anlagen-dimensionierung dienen. Einsatzbereiche der Energiesimulation sind:

- Vergleich des Jahresenergieverbrauchs verschiedener RLT – Systeme
- Optimierung der Zonen eines Lüftungssystems
- Dimensionierung von RLT – Anlagen

In einer Erweiterung der Energiesimulation kann der jährliche Gesamtenergie-verbrauch eines Gebäudes berechnet werden. Zusätzlich zum Energieverbrauch der RLT – Geräte werden auch Beleuchtung und sonstige benutzerspezifische Geräte berücksichtigt.

6.3 Simulationskern DOE-2

Dieses Kapitel gibt einen Überblick über das Simulationsprogramm DOE-2.1E (kurz DOE-2). Das Programm bildet den Simulationskern von RIUSKA, in dem die eigentliche thermische Simulation stattfindet. RIUSKA selbst dient als benutzerfreundliche Eingabeoberfläche zur einfachen Erstellung von Eingabedateien, die von DOE-2 verarbeitet werden. RIUSKA liest und „übersetzt" wiederum die Rechenergebnisse, die DOE-2 in einer Ausgabedatei bereithält.

Der Benutzer benötigt zur Bedienung von RIUSKA kein Hintergrundwissen über die Bedienung von DOE-2. Im Rahmen dieser Arbeit soll auch das Umfeld von RIUSKA und dessen Leistungsfähigkeit beleuchtet werden. Der Leistungsumfang von DOE-2 und die Rechengenauigkeit haben eine direkte Auswirkung auf die Qualität der RIUSKA – Simulationen.

6.3.1 Programmüberblick

DOE-2 wurde seit 1978 vom *Lawrence Berkeley National Laboratory* (LBNL) mit Unterstützung des amerikanischen *Department of Energy* (DOE) entwickelt. Das Programm ist frei erhältlich und richtet sich an Forschungsinstitutionen aber auch an Entwickler, da es keine graphische Eingabeoberfläche besitzt.

DOE-2 berechnet stündliche Ergebnisse im Jahresverlauf für den Energieverbrauch und die Energiekosten eines Gebäudes. Das Programm beinhaltet Rechenalgorithmen zur Verarbeitung einer Textdatei, die alle benötigten Eingabedaten enthält. Die Wetterdaten werden in Form einer Binärdatei eingelesen. Um die Handhabung von DOE-2 zu vereinfachen, wurden zahlreiche Eingabeoberflächen entwickelt (LBNL, 2006).

Der Programmablauf basiert auf der sequentiellen Abarbeitung der Subprogramme LOADS, SYSTEMS, PLANT und ECONOMICS. Die Ergebnisse der Module sind jeweils der Input für das folgende Modul. Aus dem Programmablauf ergibt sich, dass DOE-2 die Auswirkungen von Regelstrategien nicht berechnen kann, da das folgende Modul kein Feedback an seinen Vorgänger gibt.

Die Ergebnisse der Simulation werden als lesbare Textdatei ausgegeben. Für die Aufbereitung und Auswertung wurden ebenfalls Benutzeroberflächen entwickelt. Generell ist die Benutzung von DOE-2 auch ohne Benutzeroberfläche möglich, wenn auch aufwändig. Sehr gute Programmkenntnisse sind Voraussetzung.

6.3.2 Aufbau

DOE-2 ist in der Programmiersprache FORTRAN geschrieben und kann auf PC ausgeführt werden. Das Programm besteht aus fünf Unterprogrammen, die die Aufbereitung des Inputs und die eigentliche Simulation übernehmen. Abbildung 6.1 zeigt den Programmfluss einer thermischen Gebäudesimulation mit DOE-2. Die Schnittstellen zu RIUSKA sind hier bereits dargestellt und werden in Kapitel 6.4 näher erläutert.

Nach der Definierung der Eingabedaten über ein Benutzerinterface oder der Erstellung der Eingabedatei, „übersetzt" der *Building Description Language (BDL) – Prozessor* alle notwendigen Eingaben in computerlesbare Form. Weiters berechnet dieses Modul die *Responsefaktoren* für den instationären Wärmetransport durch Wände und die *Gewichtsfaktoren* zur Berechnung der zeitlichen Verzögerung durch Speichermassen (siehe auch Kapitel 6.3.4).

Die stündlichen Heiz- und Kühllasten werden im Programmteil *LOADS* berechnet. Der Berechnungsschritt ist mit einer Stunde fixiert. Die Lasten werden unter der Voraussetzung ermittelt, dass jeder Raum eine konstante Raumtemperatur besitzt. Wetterdaten, Sonneneinstrahlung sowie Größe und zeitliche Profile von inneren Lasten sind ebenso ausschlaggebend wie die natürliche Außenluftinfiltration, Wärmetransporte durch Bauteile und Beschattungen.

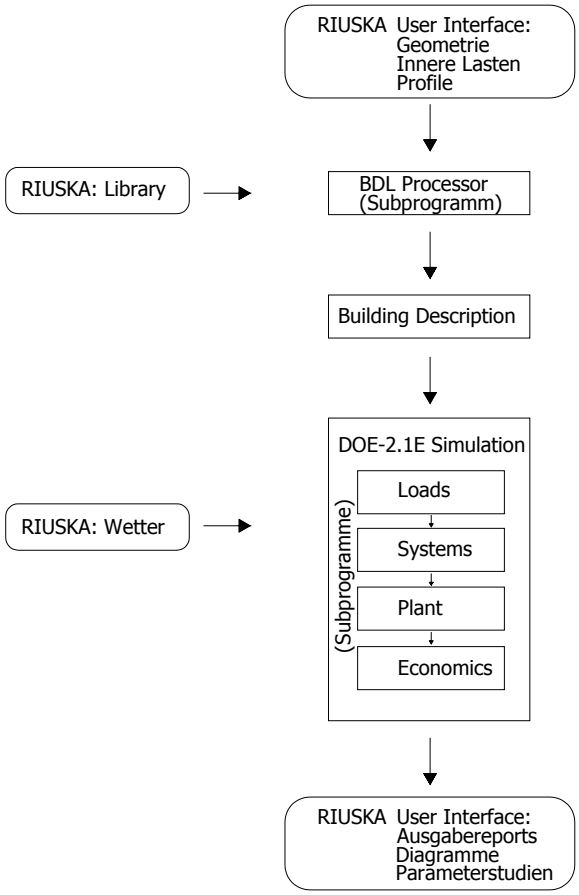


Abbildung 6.1: Programmablauf DOE-2 mit RIUSKA – Interface (nach LBNL, 2006)

Basierend auf den LOADS – Ergebnissen wird im nächsten Subprogramm *SYSTEMS* der Betrieb der lufttechnischen Anlagen simuliert. Anlagenbetriebszeiten, Regelstrategien und erforderlicher Außenluftvolumenstrom werden berücksichtigt. Luftvolumenstrom und Anlagenleistung werden von SYSTEMS ausgegeben.

Im Subprogramm *PLANT* wird das Verhalten von Anlagen (Kühler, Speicher, Heizkessel, Kühltürme etc.) mit Berücksichtigung der SYSTEMS – Ergebnisse berechnet. Für die Berechnung des Brennstoffs- und Strombedarfs wird Teillastverhalten angenommen.

Die Energiekosten werden im Programmteil *ECONOMICS* berechnet. Hier können auch Kostenvorteile unterschiedlicher Designvarianten oder Einsparungen aufgrund von Sanierungsmaßnahmen bei bestehenden Gebäuden verglichen werden.

Die Berechnungen der Module erfolgen unabhängig voneinander und sequentiell. Jedes Modul erstellt ein Ausgabefile, das für das folgende als Eingabe dient. Nach jedem Modul kann der Programmablauf unterbrochen und ein Report ausgegeben werden.

6.3.3 Wetterdaten

Grundlage jeder thermischen Simulation sind die verwendeten Wetterdaten. DOE-2 benötigt Wetterdaten in Form einer Binärdatei. Ein DOE-2 – Wetterfile enthält neben Angaben zum Standort (Längengrad, Breitengrad, Zeitzone…) folgende Datensätze:

- Trocken- und Feuchtkugeltemperatur
- Luftdruck, Luftdichte, spezifische Luftenthalpie sowie Wassergehalt der Luft
- Windgeschwindigkeit und Windrichtung
- Wolkendichte, Wolkenart (Lichtundurchlässigkeit)
- Niederschlag: Regen, Schnee (Ja/Nein)
- Solarstrahlung (Global-, Normal- und Diffusstrahlung)
- Himmelsbedeckung (monatlich)
- Erdtemperatur

Mit Hilfe des Wetterprozessors *doewth* kann aus stündlich vorliegenden Rohwetterdaten eine DOE-2 kompatible Wetterdatei erstellt werden. Die Eingabedaten müssen dem Prozessor in Form zweier ASCII – Dateien bereitgestellt werden. Eine Datei enthält stündliche Wetterdaten, die zweite Datei beschreibt den Aufbau der Ein- und Ausgabedaten. Den Programmoutput bilden zwei weitere ASCII – Dateien, wobei eine Datei den gepackten DOE-2 – Wetterdatensatz und die andere einen Statusbericht enthält (Buhl, 1999).

Die Erstellung eines DOE-2 – kompatiblen Wetterdatensatzes aus beliebigen Wetterdaten ist nicht auf „Knopfdruck" möglich. Für die oben beschriebene Prozedur müssen die Rohwetterdaten in einer der unterstützten Standardformate vorliegen. Diese Formate stammen hauptsächlich aus dem nordamerikanischen Raum (TMY2, WYEC2, TRY etc.). Messdaten anderer Formate können konvertiert werden, in dem der Dateiinhalt an die DOE-2 Spezifikationen (Anzahl und Inhalt der Spalten, Bezeichnungen…) angepasst wird. Ist eine Konvertierung der Messdaten nicht möglich, kann der Source Code von doewth erweitert werden. Fortran – Kenntnisse sind Voraussetzung. Abhängig von den vorliegenden Rohwetterdaten kann deren Aufbereitung daher sehr umfangreich sein und genaue Kenntnisse des Wetter-prozessors oder auch Programmierkenntnisse erfordern.

Um eine spätere Implementierung des Wetterdatensatzes in RIUSKA gewährleisten zu können, ist vor der Datengenerierung Rücksprache mit Granlund zu halten. So können Inhalt und Umfang der Wetterdaten auf die RIUSKA – Wetterdatenbank abgestimmt werden. Einmal generierte und in RIUSKA implementierte Datensätze sind unveränderlich und stehen für weitere Projekte im Rahmen einer Wetterbibliothek zur Verfügung.

6.3.4 Berechnungsmethoden

Im Gegensatz zur detaillierten Berechnung von Wärmeleitungsvorgängen in Wänden mittels der *Finite Elemente Methode*, verwendet DOE-2 Response- und Gewichtsfaktoren. Diese Berechnungsmethoden erfordern einen geringeren Rechenaufwand und liefern sehr reale Ergebnisse (Schinnerl et al., 2003).

Responsefaktoren beschreiben das instationäre, thermische Verhalten speicherfähiger Bauteile, indem die Reaktion auf eine Störgröße ermittelt wird. Wird ein Temperatursprung auf einer Seite der Wand aufgebracht, kann auf die Reaktion auf der anderen Wandseite geschlossen werden. Der detaillierte Temperaturverlauf in der Wand wird nicht ermittelt, wodurch die Berechnung einfacher wird und weniger Rechenkapazität erfordert. Responsefaktoren können gemessen oder berechnet werden. Faktoren von Standardbauteilen sind in einer Datenbank hinterlegt.

Analog zu Responsefaktoren werden *Gewichtsfaktoren* verwendet, um das Verhalten einer Zone bei einer Lastveränderung zu beschreiben. Für die Raumtemperatur wird ein fixer Referenzwert angenommen. Die Kühllast wird unter Berücksichtigung des Speicherverhaltens der Zone berechnet. Die verwendeten Gewichtsfaktoren hängen von der Art der Wärmegewinne ab. DOE-2 unterscheidet folgende Wärmequellen (Schinnerl, 2002): Solarstrahlung, Raumbeleuchtung, Personen und Maschinen, sowie Wärmeleitung in den Raum und Lufttemperatur.

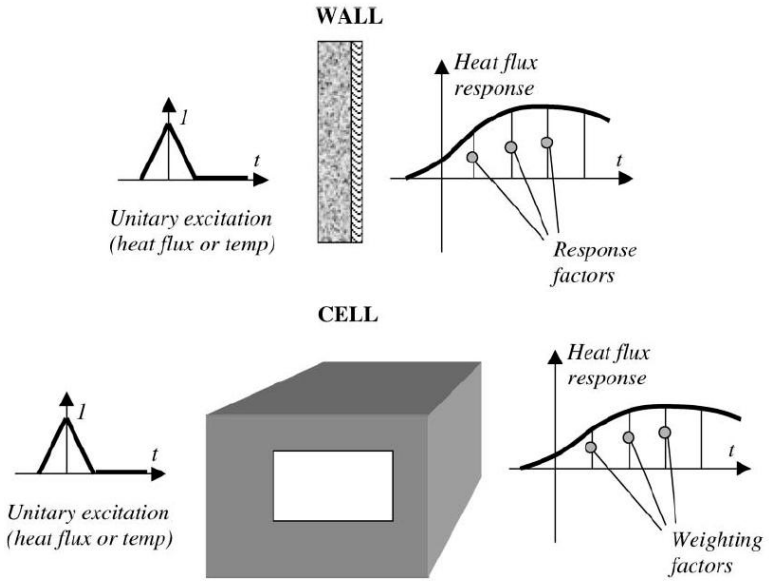


Abbildung 6.2: Response- und Gewichtsfaktoren (Depecker et al., 2001)

Depecker (2001) beschreibt die Analogie zwischen Response- und Gewichtsfaktoren (Abbildung 6.2). Auf der linken Seite ist das Aufbringen des Einheitsimpulses dargestellt. Die Antwort auf den Einheitsimpuls wird im Falle einer Wand als Response-, bei Betrachtung einer Zone (Raum oder Gebäude) als Gewichtsfaktor bezeichnet. Rechts ist der tatsächliche, zeitliche Verlauf des Wärmestroms dargestellt.

Die Funktion wird in einzelne Zeitschritte Δt zerlegt. Für jeden Zeitschritt kann ein Einheitsimpuls aufgetragen und eine Antwort in Form eines Response- bzw. Gewichtsfaktors errechnet werden. In DOE-2 ist dieser Berechnungsschritt mit einer Stunde fixiert. Das Prinzip der Response- und Gewichtsfaktorenmethode wird ausführlich von Depecker et al. (2001) erläutert. Die Berechnungsmethoden von DOE-2 werden bei Schinnerl (2002) besprochen.

6.3.5 Anwendungsbereiche

Der Einsatzbereich von DOE-2 ist auf Grund des Umfangs und des hohen Flexibilisierungsgrades der Eingabedaten sehr umfangreich und wird hier kurz umrissen (LBNL, 2006).

Anwendungsbereiche für Parameterstudien in der Energieeinsparung:

- Auswirkungen von Wandstärken, Wandaufbau, Materialauswahl und Orientierung von Außenwänden und Dächern
- Speichermasseneffekte von Wänden und Böden, sowie in Speichern der Haustechnikanlage
- Auswirkungen innerer thermischer Lasten (Personen, Beleuchtung, Anlagen) und ihrer Profile
- Auswirkungen von Unterbrechungen des Anlagenbetriebs (Nachtabsenkung…)
- Einfluss der Reduzierung des Mindestaußenluftvolumenstroms und der Nutzung von freier Kühlung
- Auswirkungen von innerer oder äußerer Beschattung, beschichteten oder reflexiven Fenstergläsern und Nutzungsmöglichkeiten des Tageslichts

Anwendungsbereiche das Gebäudedesign betreffend:

- Vorauswahl des Gebäudedesigns: Architektur, Haustechnikanlagen, Energie
- Bewertung einzelner Gebäudekonzepte hinsichtlich Raumzonierungen, Regelungsstrategien und HLK – Systemen
- Bewertung von Änderungsvorschlägen und Abweichungen des Designs während der Bauphase
- Vergleichsbasis für den gemessenen Energieverbrauch während des Betriebs
- Analyse und Bewertung von Sanierungsmöglichkeiten für bestehende Gebäude

6.3.6 Weiterentwicklung

Seit 1978 entwickelt das amerikanische DOE mit dem LBNL das Simulationsprogramm DOE-2. Parallel dazu entwickelte das amerikanische Verteidigungsministerium das Simulationsprogramm BLAST. Seit 1996 konzentriert sich die Entwicklung auf EnergyPLus, ein neues Simulationsprogramm das die Vorteile von DOE-2 und BLAST vereinigen soll und darüberhinaus weitere Anwendungsmöglichkeiten beinhaltet. EnergyPlus besitzt einen neuen Programmcode und basiert auf der Programmiersprache Fortran 90.

Die letzte offizielle Version von DOE-2 ist die Version DOE-2.1E. Trotz des Entwicklungsstopps für DOE-2 steht seitens des LBNL weiterhin Wartung und Support zur Verfügung. DOE-2, BLAST und EnergyPlus unterscheiden sich zum Teil erheblich voneinander. Die folgende Übersicht fasst die wichtigsten Features der Simulationsprogramme zusammen. Es wird zwischen allgemeinen (Tabelle 6.1), anlagenspezifischen (Tabelle 6.2) und lastspezifischen Eigenschaften (Tabelle 6.3) unterschieden. Der Vergleich ist Crawley et al. (2001) entnommen.

Tabelle 6.1: Vergleich allgemeiner Features

Feature	DOE-2	BLAST	EnergyPlus
Ganzheitlicher, simultaner Lösungsansatz ▪ Integrierte Lasten/Systeme/Anlagen ▪ Iterativer Lösungsansatz ▪ Feste Kopplung	Nein	Nein	**Ja**
Variable Berechnungszeitschritte ▪ Benutzerdefinierte Zeitschritte für Wechselwirkungen zwischen Zonen und Umwelt (15 min Voreinstellung) ▪ Variable Zeitschritte für Wechsel wirkungen Zonenluftvolumenstrom und HVAC – Anlage (≥ 1min)	Nein	Nein	**Ja**
Eingabefunktionen ▪ Benutzer kann Eingabe ohne erneute Kompilierung ändern	**Ja**	Nein	**Ja**
Ausgabefunktionen ▪ Standardberichte ▪ Benutzerdefinierte Berichte ▪ Graphische Aufbereitung	**Ja** Nein Nein	**Ja** Nein Nein	**Ja** **Ja** **Ja**

Tabelle 6.2: Vergleich von anlagenspezifischen Features

Feature	DOE-2	BLAST	EnergyPlus
Flüssigkeitsgefüllte Kreisläufe ▪ Kopplung von Energieerzeugung und Energieabgabe ▪ Kreisläufe von Heißwasser, Kühlwasser und Kondensat, Kältemittel	Nein	Nein	**Ja**
Luftführende Kreisläufe ▪ Kopplung von Ventilatoren, Auslässen, Mischkästen und Zonen	Nein	Nein	**Ja**
Benutzerdefinierte HVAC – Anlagen	Nein	Nein	**Ja**
Hochtemperaturstrahlungsheizung ▪ Gas/Elektrische Strahler, Wandstrahler	Nein	**Ja**	**Ja**
Niedertemperaturstrahlungsheizung und -kühlung	Nein	Nein	**Ja**
Kalkulation von Schadstoffemissionen ▪ CO2, SOx, NOx, CO, Partikel und Kohlenwasserstoffproduktion ▪ Bauseits und im Kraftwerk ▪ Berechnete Reduktion von Treibhaus-gasen	**Ja**	**Ja**	**Ja**
SPARK Link	Nein	Nein	**Ja**
TRNSYS Link	Nein	Nein	**Ja**

Tabelle 6.3: Vergleich von lastspezifischen Features

Feature	DOE-2	BLAST	EnergyPlus
Wärmebilanzrechnung ▪ Simultane Berechnung von Strahlungs- und Konvektionsauswirkungen für jeden Zeitschritt	Nein	**Ja**	**Ja**
Konvektion an inneren Oberflächen ▪ Abhängig von Temperatur und Luftvolumenstrom ▪ Innere, thermische Speichermassen	Nein **Ja**	**Ja** **Ja**	**Ja** **Ja**
Feuchtigkeitsabsorption/desorption ▪ Kombinierter Transport von Wärme und Masse in der Gebäudehülle	Nein	Nein	**Ja**
Thermischer Komfort ▪ Menschliches Komfortmodell basierend auf Aktivitätsgrad, innerer Trockenkugeltemperatur, Feuchtigkeit und Strahlung	Nein	**Ja**	**Ja**
Anisotropes Himmelsmodell ▪ Himmelsstrahlung hängt vom Sonnenstand ab, für eine bessere Berechnung der diffusen Strahlung auf geneigten Flächen	**Ja**	Nein	**Ja**
Fortgeschrittene Fensterberechnungen ▪ Kontrollierbare Fensterläden ▪ Elektrochromatische Verglasung	**Ja**	Nein	**Ja**
„WINDOW 5“ Berechnungen ▪ Mehr als 200 Fensterarten – konventionell, verspiegelt, Low-E, gasgefüllt, elektrochromatisch ▪ Schicht – für – Schicht Eingabe von benutzerdefinierter Verglasung	**Ja** Nein	Nein Nein	**Ja** **Ja**
Tageslichtsimulation und -regelung ▪ Innenbeleuchtung durch Fenster und Dachfenster ▪ Stufen, Dimmer, An/Aus Beleuchtungsregelung ▪ Blendungssimulation und -regelung ▪ Auswirkungen von Dimmern auf Heizung und Kühlung	**Ja**	Nein	**Ja**

6.4 DOE-2 und Riuska

6.4.1 DOE-2 in RIUSKA

Das Konzept von DOE-2 basiert auf der Bereitstellung eines Analyseprogramms für den Energieverbrauch eines Gebäudes, sowie die wirtschaftliche Betrachtung von Energiekosten. Die Berechnung der thermischen Bedingungen im Gebäude ist ebenfalls möglich. DOE-2 ist public domain, d.h. es ist frei zugänglich. Die Erstellung von benutzerfreundlichen Eingabeoberflächen ist ausdrücklich erwünscht. Ein Simulationsprogramm, das die Rechenalgorithmen von DOE-2 für die Auswahl und Dimensionierung von haustechnischen Anlagen nutzt, ist RIUSKA.

DOE-2 wurde als Grundlage für RIUSKA gewählt, da es für Anwendungen in der Gebäude- und Energiesimulation hinsichtlich Genauigkeit und Berechnungszeiten optimiert wurde. Die verfolgten Entwicklungsstrategien und der Support für das Programm entsprechen den RIUSKA Strategien (Jokela et al, 1997). Zukünftig planen die Entwickler den Einsatz von EnergyPlus als Simulationskern für RIUSKA. Ein Zeitplan für die Implementierung und der endgültige Umfang der Features stehen noch nicht fest (Karola, 2006).

In RIUSKA wird DOE-2 zur Energiesimulation des Gebäudes und zur Behaglichkeitssimulation einzelner Räume verwendet. Bei der Energiesimulation werden die Heiz- und Kühlleistungen der Anlagen ermittelt. Der Energieverbrauch der Ventilatoren wird von RIUSKA berechnet.

Für die Energiesimulation wird eine binäre Wetterdatei benötigt, die 8760 Stundenwerte für das gesamte Jahr enthält. DOE-2 kompatible Wetterdaten können von den Entwicklern von DOE-2 (LBNL) oder von Olof Granlund Oy bezogen werden. Die Behaglichkeitssimulation dient zur Auslegung von Anlagen bzw. Volumenströmen am *Design Day*, dem Auslegungstag. Für die Simulation werden die Wetterdaten des Design Days benötigt.

Die Materialdatenbank von RIUSKA enthält Fensterparameter, die nicht geändert werden können. Fensterdatensätze, die mit DOE-2 kompatibel sind, werden speziell erstellt und in die RIUSKA – Materialdatenbank importiert. Fensterdatensätze können bei Olof Granlund Oy angefordert werden.

RIUSKA erstellt eine Eingabedatei für DOE-2, die vor der Simulation über einen Editor bearbeitet werden kann. Werden der Input – Datei neue Variablen hinzugefügt oder Zahlenwerte über ein angemessenes Limit hinaus verändert, kann es zu Fehlern bei der Simulation kommen. Fehler treten besonders dann auf, wenn RIUSKA Daten aus der Output – Datei in die Datenbank einlesen soll.

Die Outputdateien von DOE-2 der zuletzt durchgeführten Simulation können als Textdatei abgerufen werden. Der Inhalt wird in der RIUSKA – Datenbank gespeichert und ist auch nach Durchführung einer weiteren Simulation zugänglich, während die Textdatei überschrieben wird. DOE-2 gibt Ergebnisse in Form einer Textdatei aus. Da RIUSKA nur Teile von DOE-2 unterstützt, werden nicht alle verfügbaren Berichte

erstellt. Im Rahmen der Energiesimulation werden in RIUSKA anlagenunabhängige Leistungen, z. B. der Jahresverlauf der inneren, thermischen Lasten, nicht ausgegeben. Die folgenden Abbildungen zeigen Auszüge aus den DOE – Outputdateien.

REPORT- SV-A SYSTEM DESIGN PARAMETERS AC_1 WEATHER FILE- Hamburg 19

SYSTEM NAME	SYSTEM TYPE	ALTITUDE MULTIPLIER	FLOOR AREA (M2)	MAX PEOPLE
AC_1	RHFS	1.000	907.4	20.

SUPPLY FAN (M3/H)	ELEC (KW)	DELTA-T (C)	RETURN FAN (M3/H)	ELEC (KW)	DELTA-T (C)	OUTSIDE AIR RATIO	COOLING CAPACITY (KW)	SENSIBLE (SHR)	HEATING CAPACITY (KW)	COOLING EIR (KWH/KWH)	HEATING EIR (KWH/KWH)
12667.	7.530	1.0	0.	0.000	0.0	1.000	10000.000	0.524	-10000.000	0.00	0.37

ZONE NAME	SUPPLY FLOW (M3/H)	EXHAUST FLOW (M3/H)	FAN (KW)	MINIMUM FLOW RATIO	OUTSIDE AIR FLOW (M3/H)	COOLING CAPACITY (KW)	SENSIBLE (SHR)	EXTRACTION RATE (KW)	HEATING CAPACITY (KW)	ADDITION RATE (KW)	MULTIPLIER
Spc1	6118.	0.	0.000	1.000	6118.	0.00	0.00	11.94	0.00	-9990.56	1.0
Spc2	6094.	0.	0.000	1.000	6094.	0.00	0.00	11.89	0.00	-9990.59	1.0
Spc3	456.	0.	0.000	1.000	456.	0.00	0.00	1.63	0.00	-10002.64	1.0

Abbildung 6.3: DOE – Output einer Behaglichkeitssimulation (Auszug)

Form und Inhalt der Ergebnisberichte werden ausführlich im DOE – Manual beschrieben. Die in den Outputdateien ermittelten Werte werden in den RIUSKA – Berichten benutzerfreundlich aufbereitet.

REPORT- LS-D BUILDING MONTHLY LOADS SUMMARY WEATHER FILE- Hamburg 19

	COOLING					HEATING					ELEC	
MONTH	COOLING ENERGY (MWH)	TIME OF MAX DY	TIME OF MAX HR	DRY-BULB TEMP	WET-BULB TEMP	MAXIMUM COOLING LOAD (KW)	HEATING ENERGY (MWH)	TIME OF MAX DY	TIME OF MAX HR	DRY-BULB TEMP	WET-BULB TEMP	MAXIMUM HEATING LOAD (KW)	ELEC-TRICAL ENERGY (KWH)	MAXIMUM ELEC LOAD (KW)
JAN	0.00035	26	14	13.C	11.C	0.144	-7.752	7	7	-11.C	-11.C	-21.287	9373.	12.598
FEB	0.02185	22	15	13.C	9.C	3.401	-6.884	4	1	-8.C	-8.C	-20.089	8466.	12.598
MAR	0.07038	27	13	17.C	12.C	5.411	-5.903	10	7	-10.C	-11.C	-17.297	9373.	12.598
APR	0.69850	19	12	13.C	10.C	11.843	-3.127	7	5	-2.C	-3.C	-11.355	9071.	12.598
MAY	2.57815	25	13	19.C	13.C	16.612	-0.713	10	5	1.C	-1.C	-7.982	9373.	12.598
JUN	4.61215	24	14	18.C	13.C	20.269	-0.072	9	4	6.C	6.C	-2.976	9071.	12.598
JUL	5.94493	17	14	23.C	17.C	24.349	-0.030	26	3	5.C	5.C	-2.318	9373.	12.598
AUG	6.12291	6	15	23.C	16.C	25.528	-0.016	16	4	7.C	7.C	-2.028	9373.	12.598
SEP	3.71720	20	15	27.C	21.C	19.092	-0.083	4	6	4.C	4.C	-4.212	9071.	12.598
OCT	0.90566	8	14	23.C	18.C	10.632	-0.969	17	4	1.C	1.C	-8.281	9373.	12.598
NOV	0.03205	27	14	17.C	14.C	2.881	-3.879	15	7	-6.C	-6.C	-13.663	9071.	12.598
DEC	0.00000					0.000	-6.473	13	7	-9.C	-9.C	-17.766	9373.	12.598
TOTAL	24.704						-35.900						110361.	
MAX						25.528						-21.287		12.598

Abbildung 6.4: DOE – Output der Energiesimulation (Auszug)

6.4.2 DOE-2 ohne RIUSKA

Die Simulation kann auch ohne das RIUSKA – Interface durchgeführt werden. Die Input – Datei kann im Editor geöffnet und bearbeitet werden. Für die Verwendung von DOE-2 ohne RIUSKA werden genaue Programmkenntnisse empfohlen. In diesem Zusammenhang sei auf die Online – Hilfe von RIUSKA bzw. das im Zuge dieser Arbeit erstellte Benutzerhandbuch für RIUSKA sowie das DOE-2 Manual des LBNL verwiesen.

6.5 IFC – Schnittstelle

In Version 4.0.13 ist RIUSKA kompatibel zu IFC2x2 und älteren Versionen. Das Programm generiert selbst keine IFC – Dateien. Über die Vermittlungssoftware BSPro COM Server können das Gebäudemodell importiert und Simulationsergebnisse exportiert werden.

Das Programm ist Client des Middleware Tools BSPro COM – Server. Durch den Einsatz von BSPro COM – Server verringert sich der Aufwand zur Übernahme des Gebäudemodells erheblich. Die importierte Geometrie wird auf die Bedürfnisse der Simulation abgestimmt. Berechnungsergebnisse der RIUSKA – Simulationen können über BSPro COM in die IFC – Datei exportiert werden. (Karola et al, 2001). BSPro COM Server ist im Softwarepaket von RIUSKA inkludiert. Das Programm wird automatisch beim Modellimport ausgeführt. Für den Anwender entsteht kein Mehraufwand.

RIUSKA – Simulationen basieren ausschließlich auf bestehenden IFC – Gebäude-modellen. Das Programm bietet keine Möglichkeit zur Änderung des Modells. Das Modell muss vor der Verarbeitung in einem IFC – kompatiblen CAD – Programm überprüft und falls nötig bearbeitet werden.

Aus dem IFC – Modell übernimmt RIUSKA Grundrisse, Umschließungsflächen, Lage und Größe von Fenstern und Türen sowie Raum- und Bauteilbezeichnungen. Ein selektiver Import einzelner Geschoße des Gebäudes ist möglich. Umfang und Qualität der IFC – Importe werden im Praxistest (Kapitel 7) eingehend erläutert.

Auf Grund der verwendeten Schnittstelle importiert RIUSKA zur Zeit keine Wandauf-bauten oder u – Werte, auch wenn diese im IFC – Modell enthalten sind. Auf den Import von Wandaufbauten wird verzichtet, da in vielen IFC – Dateien die Bauteile falsch oder gar nicht definiert sind (Karola, 2006).

Ergebnisse der Raumsimulation können in das Gebäudemodell der IFC zurück-geschrieben werden. Die Ergebnisse sind u. a. in Planungsprogrammen zur Dimensionierung des Kanalnetzes verwendbar. In das IFC – Modell übertragen werden:

- Luftvolumenstrom (unter der Voraussetzung: Zuluft = Abluft)
- zusätzliche Kühllast
- Auslegungstemperatur Kühlfall
- Auslegungstemperatur Heizfall
- Zonenaufteilung

Die praktische Handhabung der Übertragung von Informationen in das IFC – Modell wird in Kapitel 7.5.3 behandelt.

6.6 Thermische Simulationen in RIUSKA

In RIUSKA stehen für die thermische Simulation des Gebäudes zwei verschiedene Berechnungsarten zur Verfügung: die Behaglichkeitssimulation (oder Raumsimulation) und die Energiesimulation (oder Anlagensimulation). Abbildung 6.5 zeigt die möglichen Simulationsarten.

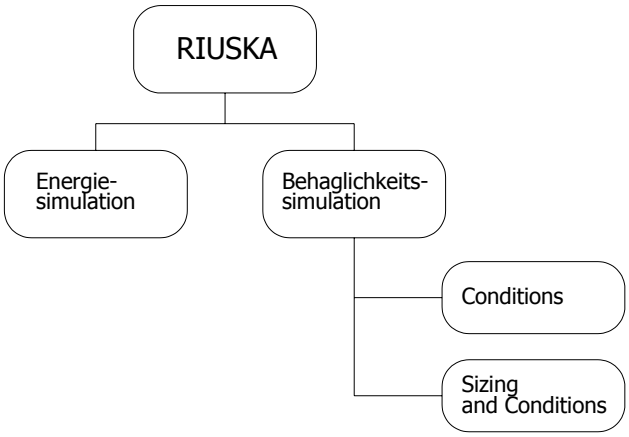


Abbildung 6.5: Simulationsmöglichkeiten in RIUSKA

Bei der Durchführung einer Energiesimulation wird der gesamte jährliche Energieverbrauch für Heizung, Kühlung und Ventilation des Gebäudes berechnet. Zusätzlich kann der Gesamtenergieverbrauch berechnet werden, wenn Angaben über die elektrischen Anlagen im Gebäude gemacht werden. Die Energiesimulation basiert auf Eingabedaten der Raumdatenbank oder auf einer *Sizing and Conditions* – Simulation. Ergebnis der Energiesimulation sind der jährliche Energieverbrauch und der Gesamtluftvolumenstrom sowie die Gesamtkühlleistung am Auslegungstag.

Bei der Behaglichkeitssimulation wird auf Grund von Geometrie, Wetterdaten, inneren Lasten und Profilen sowie Angaben zu Innentemperatur und Mindestluftvolumenstrom der Zustand in einem einzigen Raum am Auslegungstag berechnet. Im Zuge der Simulation können die Behaglichkeitsindices PMV und PPD berechnet werden. Die Behaglichkeitssimulation bietet zwei Berechnungsmöglichkeiten:

- *Conditions*: Der Temperaturverlauf wird bei festgelegtem Luftvolumenstrom und Kühlleistung berechnet. Diese Option ermöglicht es festzustellen, ob ausgewählte Anlagen tatsächlich Behaglichkeit im Raum gewährleisten können.

- *Sizing and Conditions*: Luftvolumenstrom und Kühlleistung werden so berechnet, dass eine bestimmte Innentemperatur im Raum gewährleistet werden kann. Nach der iterativen Ermittlung des Luftvolumenstroms, wird automatisch eine Conditions – Simulation durchgeführt, um die Behaglichkeit im Raum zu bestätigen.

Im Gegensatz zum jährlichen Verlauf der Innentemperatur wird der Verlauf der thermischen Last in einem Raum nur für den Auslegungstag ausgegeben. Im Jahresverlauf ermittelt werden anlagenabhängige Kühl- und Heizleistungen für jeden Raum.

Werden einzelne Räume aus dem Projekt gelöscht, hat dies keine Auswirkung auf die Simulation benachbarter Räume. Der Wärmestrom zwischen Räumen wird in RIUSKA nicht berücksichtigt. Es werden gleiche Innentemperaturen angenommen. Abweichende Räume müssen extra berücksichtigt werden. Für die Simulation sind daher nicht die umgebenden Räume ausschlaggebend, sondern die Zuweisung von Außen- und Innenwänden zu einem Raum.

7 IFC – Modelle in der Praxis

Dieses Kapitel fasst praktische Erfahrungen mit IFC – Modellen zusammen. Betrachtet wird die Weiterverwendung von IFC – Modellen, die von CAAD – Programmen generiert wurden. Die Erstellung einfacher IFC – Modelle in der Haustechnik – CAD – Anwendung DDS bildet einen zweiten Schwerpunkt. Alle verwendeten Modelle werden mit dem Analyseprogamm IfcStoreyView des Forschungszentrums Karlsruhe überprüft. Anschließend werden die IFC – Modelle in DDS und RIUSKA eingelesen. Die Qualität der Modelle für die Verwendung in der haustechnischen Planung und der Gebäudesimulation mit RIUSKA wird beurteilt.

7.1 Beispielprojekte der IAI

Im Rahmen des Anwenderhandbuchs (siehe Kapitel 4.7.2) stellt die IAI fünf IFC – Gebäudemodelle online zur Verfügung. Alle Modelle enthalten ein Einfamilienhaus gleichen Umfangs und Aufbaus. Die Modelle liegen in den IFC – Formaten 2x und 2x2 vor. Die Modellerstellung fand in Zusammenarbeit mit den jeweiligen CAAD – Programmherstellern statt (IAI, 2006d). Software folgender Hersteller wurde für die Modellgenerierung verwendet:

- Autodesk: Architectural Desktop 2006 (ADT)
- Graphisoft: ArchiCAD 9
- Nemetschek: ALLPLAN 2006.0a
- Bentley: Architecture (auf Basis von Microstation 2004)
- Autodesk: Revit Building 9.1

Diese fünf IFC – Modelle der IAI werden für den Praxistest verwendet. Sie sind sozusagen vom „Architekten" zur Verfügung gestellt und bilden die Grundlage der weiteren Planung durch den Gebäudetechniker.

Wandaufbauten werden in den Modellen nicht übertragen, können aber in Form von PSETs enthalten sein. Architektonische Elemente wie Balkongeländer, Treppen und Fassadenelemente sind in einigen Modellen enthalten. Die Übertragung dieser Objekte spielt hier keine Rolle, da die IFC – Schnittstelle von RIUSKA sie nicht überträgt. Der Versuch überprüft, ob die von RIUSKA erkennbaren Elemente tatsächlich übertragen werden. Aus den IFC – Modellen sollen

- Grundrisse,
- Umschließungsflächen,
- Räumliche Lage und Ausdehnung von Wandöffnungen,
- Bezeichnung aller Bauelemente (Wände, Fenster, Türen) sowie
- Bezeichnungen und Nummerierung von Geschoßen und Räumen übertragen werden.

Geschoßdecken und Dächer werden nicht übertragen sondern gegen RIUSKA – Standardbauteile ausgetauscht.

Die Abbildungen 7.1 bis 7.4 zeigen die Grundrisse an denen sich alle Testmodelle orientieren. In Raumaufteilung, Anzahl und Größe der Fenster und Türen sowie Geschoßhöhen unterscheiden sich die Modelle geringfügig. Raumbezeichnungen sind nicht eingetragen, da sie von den Modellerstellern individuell vergeben wurden. Im Erdgeschoß ist jener Raum schraffiert, der für das Beispiel in Kapitel 7.5.3 verwendet wird.

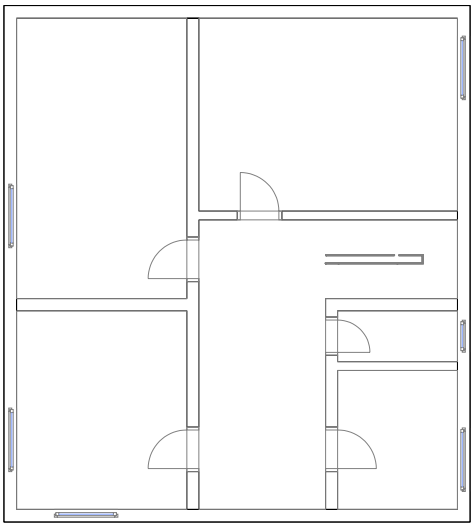


Abbildung 7.1: Kellergeschoß (IAI, 2006d)

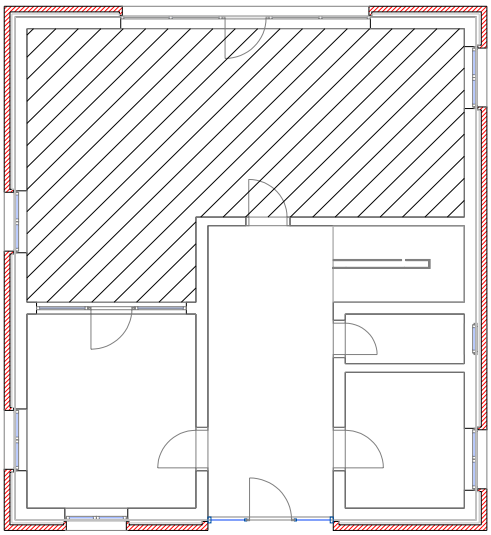


Abbildung 7.2: Ergeschoß (IAI, 2006d)

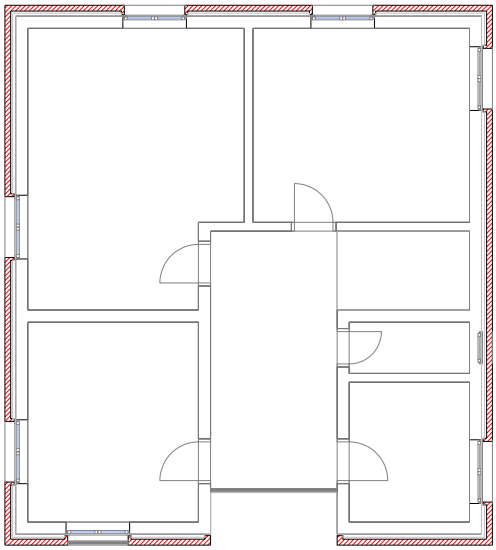


Abbildung 7.3: Obergeschoß (IAI, 2006d)

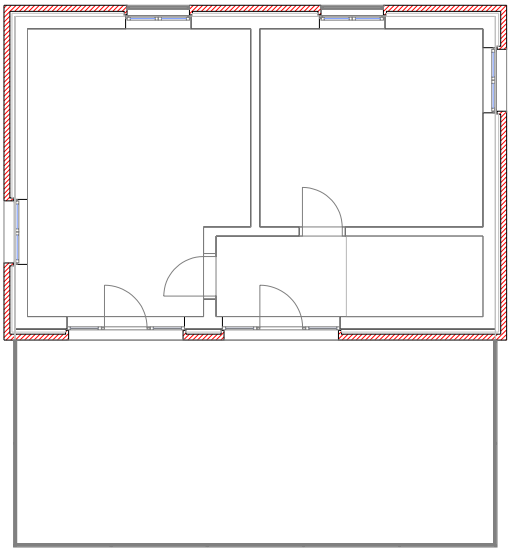


Abbildung 7.4 Dachgeschoß (IAI, 2006d)

7.2 Überprüfung der IFC – Modelle

Um die Modelle schon vor dem Import in RIUSKA vergleichen und eventuelle Fehler erkennen zu können, werden sie in IfcStoreyView überprüft. ADT und ALLPLAN verwenden das IFC – Format 2x. Die Modelle von ArchiCAD, Bentley und Revit nutzen die aktuelle IFC – Schnittstelle 2x2.

Alle Modelle enthalten Informationen (u – Werte für Fenster, Materialbezeichnungen, Zeichnungseigenschaften), die in sogenannten PSETs übertragen werden. PSETs werden in nachfolgenden Programmen nicht direkt verarbeitet. Sie sind eine in tabellenform an das Objekt angehängte Zusatzinformation.

Abbildung 7.5 bis Abbildung 7.9 zeigen für jedes IFC – Modell die dreidimensionale Ansicht im IfcStoreyView. Bei der Analyse in diesem Programm wird das Augenmerk auf folgende Punkte gelegt:

- Inhalt des Modells (in Form einer statistischen Übersicht)
- Hierarchischer Aufbau des Modells (Geschoße, Räume)
- Vollständigkeit der 2D- und 3D Darstellung
- Anzahl der enthaltenen Geschoße

In der Statistik des IfcStoreyView werden die Hauptobjekte des Modells geführt: Projekt, Gebäude, Stockwerke, Räume…. Um die Qualität des IFC – Imports in RIUSKA beurteilen zu können, sind die Anzahl der Stockwerke, Räume, Fenster und Türen ausschlaggebend. Die Anzahl der Wände wird nicht betrachtet, da in RIUSKA Trennwände zwischen Räumen jedem Raum als eigenständige Wand zugewiesen werden. Die Zahl der in RIUSKA enthaltenen Wände unterscheidet sich somit immer vom ursprünglichen Modell.

Das ADT – Modell (Abbildung 7.5) enthält 5 Stockwerke, wobei dem Dach ein eigenes Stockwerk zugewiesen wird. 21 Räume verteilen sich auf die vier verbleibenden Stockwerke. Das Gebäude besitzt 28 Fenster und 16 Türen. Das Modell wird fehlerfrei dargestellt.

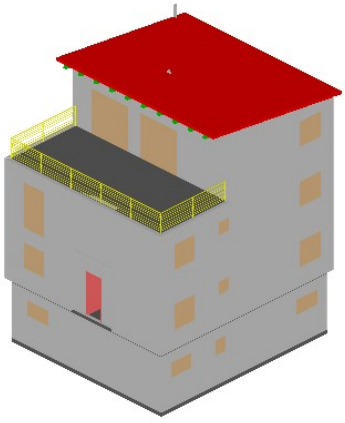


Abbildung 7.5: IFC – Modell von ADT

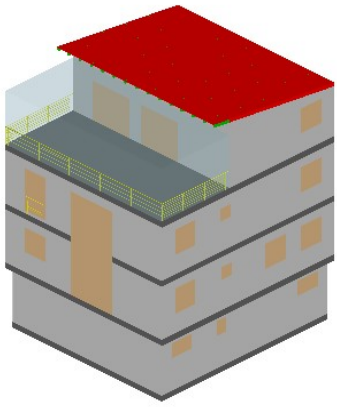


Abbildung 7.6: IFC – Modell von ArchiCAD

Ebenfalls fehlerfrei ist das ArchiCAD – Modell (Abbildung 7.6). Es enthält 4 Stockwerke mit insgesamt 21 Räumen. Das Gebäude besitzt 30 Fenster und 22 Türen. In der dreidimensionalen Ansicht ist zu erkennen, dass die Dachterrasse grau hinterlegt ist. Ihr ist ein eigener Raum zugewiesen.

In Abbildung 7.7 ist das ALLPLAN – Modell dargestellt. Es enthält vier Stockwerke. Die insgesamt 23 Räume besitzen 27 Fenster und 17 Türen. Das Modell enthält zusätzlich eine Fundamentplatte. IfcStoreyView erkennt keine Fehler im Modell.

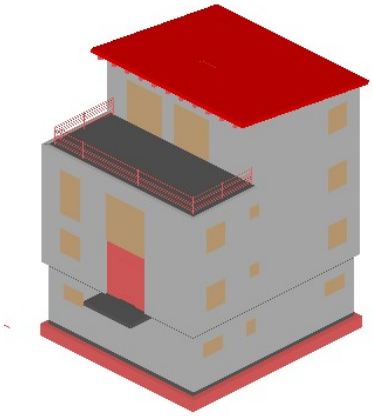


Abbildung 7.7 IFC – Modell von ALLPLAN

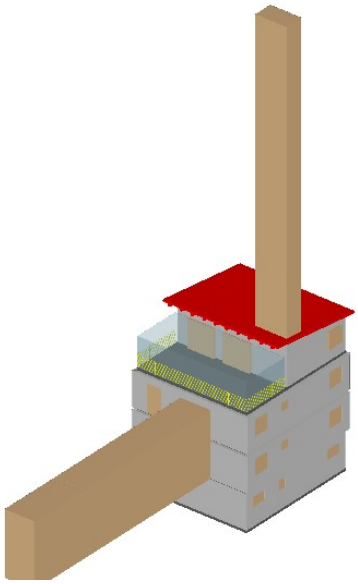


Abbildung 7.8: IFC – Modell von Bentley

Abbildung 7.8 zeigt die 3D – Ansicht des Bentley – Modells im IfcStoreyView. Elemente der Decken- bzw. Mauerkonstruktion im Erd- und Obergeschoß werden stark vergrößert dargestellt. Das Modell wird unverändert für die RIUSKA – Importe übernommen, da ein Fehler in der IfcStoreyView – Darstellung nicht ausgeschlossen werden kann. Das Modell enthält vier Stockwerke mit insgesamt 21 Räumen, die 25 Fenster und 18 Türen besitzen. Die Dachterrasse wird als eigenständiger Raum geführt.

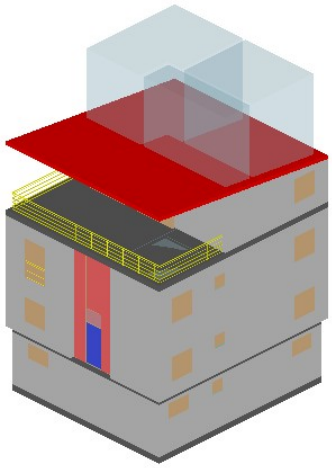


Abbildung 7.9: IFC – Modell von Revit

Das Revit – Modell (Abbildung 7.9) besitzt fünf Stockwerke, wobei dem Dach ein eigenes Stockwerk zugewiesen wird. In den vier regulären Stockwerken befinden sich 20 Räume mit 24 Fenstern und 22 Türen. Die Räume des Dachgeschoßes ragen über das Gebäude hinaus. Die Raumhöhe der IfcSpaces entspricht nicht der Geometrie des übrigen Gebäudes.

Die erste Überprüfung der von der IAI zur Verfügung gestellten IFC – Modelle ergibt, dass alle Modelle – trotz einiger Abweichungen in der dreidimensionalen Darstellung – keine schwerwiegenden, funktionsbeeinträchtigenden Fehler aufweisen. Die Modelle werden für den Praxistest unverändert übernommen.

7.3 Import der IFC – Modelle in DDS

In diesem Teil des Praxistests werden die Gebäudemodelle der IAI in die Haustechnikapplikation Data Design System: SHK Partner 6.35 (kurz DDS) eingelesen. Die Modelle bilden in DDS die Grundlage der haustechnischen Planung. Änderungen in der Architektur können nicht vorgenommen werden.

Für alle Modelle werden einheitliche Importeinstellungen verwendet. Aus den importierten Modellen wird in DDS ein programmeigenes Gebäudemodell erstellt. Anschließend werden die Modelle auf folgende Punkte hin untersucht:

- Vollständigkeit
- Anzahl der Geschoße
- Benennung der Geschoße in der Zeichnungsliste
- Korrekte Darstellung von architektonischen Bauteilen
- Darstellung des Gebäudemodells
- Darstellung von Einrichtung (z. B. Treppenhaus)
- Übertragung von Bauteileigenschaften
- Inhalt der IfcProperties (z. B. u – Werte)

Der Grundriss gilt als korrekt übertragen, wenn sowohl Raumaufteilung, Lage und Größe von Fenster und Türen als auch die Wandeigenschaften (außen- oder innenliegend) mit dem IFC – Modell übereinstimmen. Die korrekte Raumhöhe muss sowohl in der geometrischen Darstellung im Gebäudemodell als auch in der Raumdatenbank dem ursprünglichen Modell entsprechen.

Abbildung 7.10 stellt das Gebäudemodell dar, das nach Import der IFC – Datei von ADT erstellt wurde. In DDS sind 5 Geschoße enthalten, wobei das zweite Obergeschoß keine Räume besitzt und das oberste Geschoß unbenannt ist. Das Dachgeschoß wird 2,55 m über dem Erdgeschoß angeordnet. Das Dach wird in der 3D – Darstellung des Gebäudemodells nicht angezeigt.

Den Grundriss des Erdgeschoßes zeigt Abbildung 7.11. Die Außenwände werden als doppelte Innenwände dargestellt und überschneiden sich an den Eckpunkten des Gebäudes. Die Raumhöhe im Gebäudemodell entspricht dem ursprünglichen IFC – Modell. In der Raumliste wird davon abweichend eine Raumhöhe von 0,25 m angezeigt. In PSETs sind u – Werte für Wände und zum Teil auch Fenster hinterlegt.

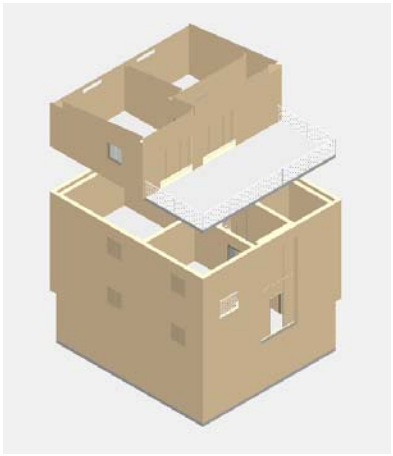


Abbildung 7.10: ADT – Modell in DDS

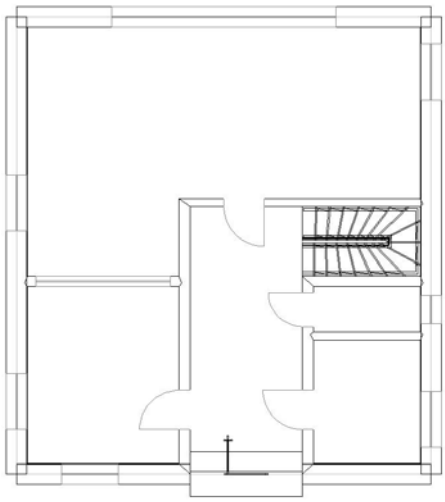


Abbildung 7.11: EG – Grundriss (ADT)

Im Rendering, d. h. in der photo-realistischen dreidimensionalen Darstellung des Gebäudemodells, zeigen sich teilweise erhebliche Abweichungen nach dem Programmneustart von DDS. Abbildung 7.12 zeigt die Render – Darstellung des ADT – Modells nach einem erneuten Import sowie einem Programmneustart. Die Importoptionen sind dabei unverändert.

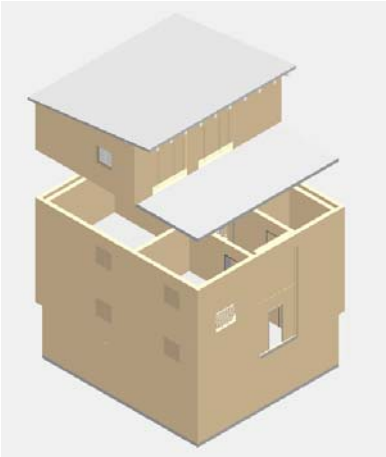


Abbildung 7.12: ADT – Modell in DDS (2)

Beim Import des ArchiCAD – Modells fielen keine Besonderheiten auf. Alle Grundrisse werden korrekt übernommen und in der Raumliste bezeichnet. Die Raumhöhen stimmen mit dem IFC – Modell überein. In PSETs sind u – Werte für Wände enthalten. Die Farbdarstellung des 3D – Gebäudemodells variiert bei verschiedenen Importversuchen. Trotz veränderlichen Farben werden im Rendering stets alle Objekte dargestellt. Die Abbildungen 7.13 und 7.14 zeigen das Gebäudemodell im Rendering – Modus sowie den Grundriss des Erdgeschoßes.

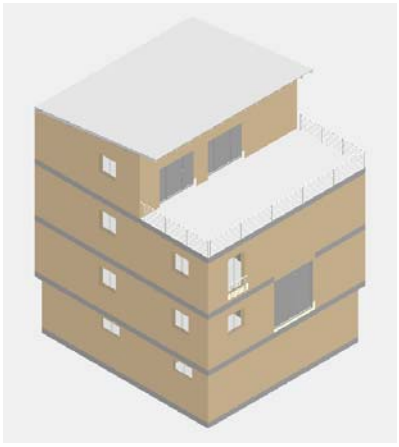


Abbildung 7.13: ArchiCAD – Modell in DDS

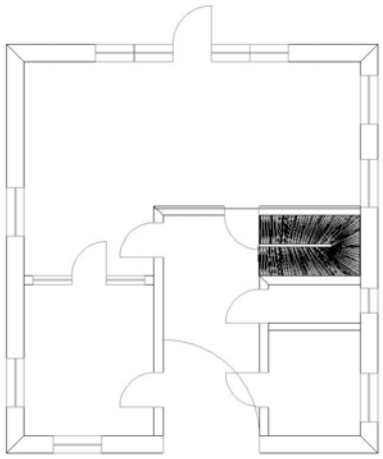


Abbildung 7.14: EG – Grundriss (ArchiCAD)

Abbildung 7.15 zeigt das Gebäudemodell, Abbildung 7.16 den Grundriss des Erdgeschoßes des von ALLPLAN erstellten IFC – Modells. Grundrisse, Raumhöhen sowie Geschoßbezeichnungen werden in DDS korrekt dargestellt. u – Werte sind für Wände im Rahmen der PSETs enthalten. Im Farbrendering fehlen Fenster des Erd-, Ober- und Dachgeschoßes. Die Darstellung bleibt bei mehreren Importversuchen und Programmneustarten unverändert.

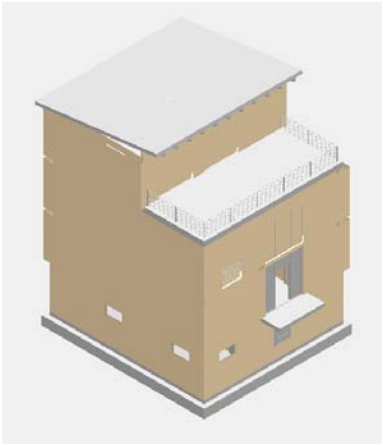


Abbildung 7.15: ALLPLAN – Modell in DDS

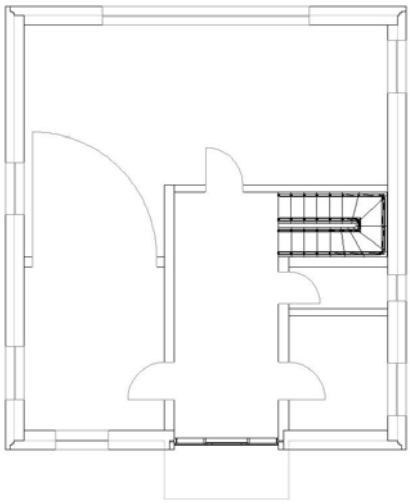


Abbildung 7.16: EG – Grundriss (ALLPLAN)

Ergebnisse des Bentley – Modellimports sind in den Abbildungen 7.17 und 7.18 dargestellt. Grundrisse, Raumhöhen und Geschoßbezeichnungen werden korrekt übertragen. In den PSETs sind anstelle von u – Werten genaue Schichtaufbauten der Wände angegeben. In der photorealistischen Darstellung werden Sparren der Dachkonstruktion stark vergrößert dargestellt.

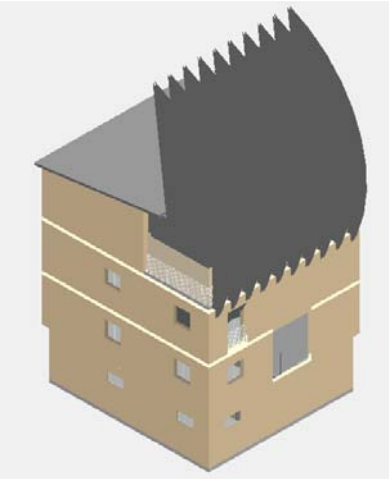


Abbildung 7.17: Bentley – Modell in DDS

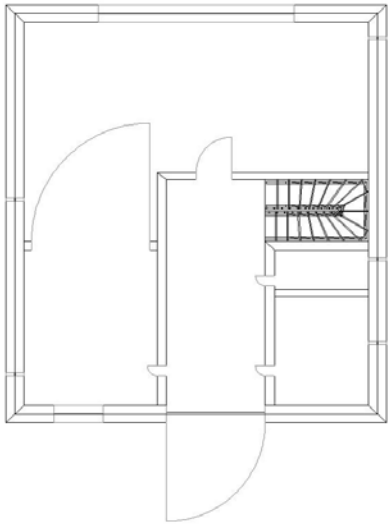


Abbildung 7.18: EG – Grundriss (Bentley)

Das Revit – Modell ist in den nachstehenden Abbildungen 7.19 und 7.20 dargestellt. Grundrisse, Raumhöhen und Geschoßbezeichnungen werden korrekt übernommen. PSETs enthalten u – Werte für Wände. In der photorealistischen Darstellung fehlen die oberste Geschoßdecke und das Dach.

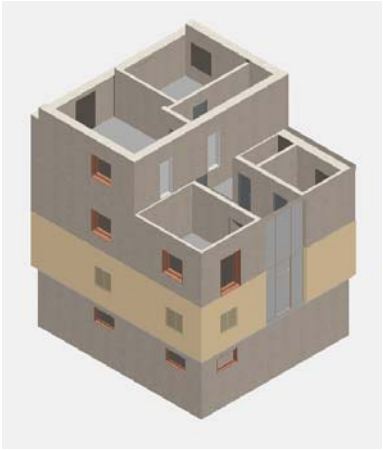


Abbildung 7.19: Revit – Modell in DDS

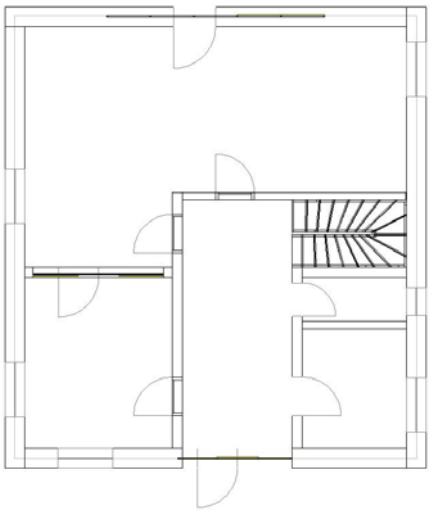


Abbildung 7.20: EG – Grundriss (Revit)

Nach einem neuerlichen Import, bei gleichbleibenden Einstellungen, ändert sich die 3D – Darstellung des Gebäudemodells (Abbildung 7.21). Obergeschoß und Dach werden nicht dargestellt. Auch die Farbdarstellung der Wände variiert. Grundrisse, Raumhöhen und Geschoßbezeichnungen weichen jedoch nicht vom ursprünglichen Import ab.

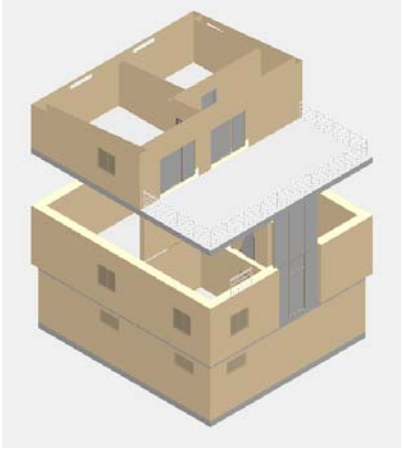


Abbildung 7.21: Revit – Modell in DDS (2)

DDS bietet die Möglichkeit importierte IFC – Objekte (Wände, Fenster, Türen) gegen Artikel der programmeigenen Datenbank zu tauschen. Diese Möglichkeit ist für alle getesteten Modelle möglich. Der Fußboden wird in keinem Modell als IFC – Objekt übertragen. Den Geschoßen ist daher von Beginn an ein Standardfußboden aus der DDS – Artikeldatenbank zugewiesen.

Tabelle 7.1 fasst die Ergebnisse des Imports der IAI – Modelle in DDS zusammen. Der Import des ADT – Modell weist die größten Mängel auf. Das Obergeschoß wird nicht importiert. Außenwände der importierten Geschoße werden als Innenwände interpretiert. Sowohl in der Zeichnungsliste als auch in der DDS – Raumdatenbank sind fehlerhafte Bezeichnungen zu finden. Der Import der weiteren Modelle weist in der Grundrissansicht, der Zeichnungsliste und den Raumdatenbanken keine Fehler auf. Die photorealistische Darstellung ist bei allen Modellen mit Ausnahme von ArchiCAD fehlerhaft. Zudem variiert sie bei den Modellen von ADT, ArchiCAD und Revit nach einem Programmneustart oder einem erneuten Import des betreffenden Modells.

Tabelle 7.1: Ergebnisse des Imports der IAI – Modelle in DDS

Programm	Korrekter Grundriss	Korrekte Raumhöhe	u – Werte in PSETs	Geschoß-bezeichnungen	Darstellung im Rendering
ADT	Nein	Nein	Wände, z. T. Fenster	1 Geschoß unbenannt	variiert; OG fehlt
ArchiCAD	Ja	Ja	Wände	vorhanden	variiert
ALLPLAN	Ja	Ja	Wände	vorhanden	Fenster fehlen
Bentley	Ja	Ja	nur Wandaufbauten	vorhanden	Sparren stark vergrößert
Revit	Ja	Ja	Wände	vorhanden	variiert

7.4 Erstellung eines neuen IFC – Modells in DDS

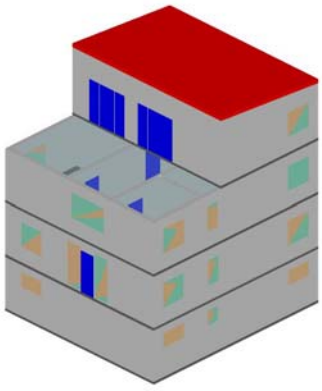


Abbildung 7.22: Ansicht des DDS – Modells in IfcStoreyView

Zusätzlich zu den IFC – Modellen der IAI wird ein Einfamilienhaus gleichen Aufbaus in DDS (Version 6.35) erstellt. Die dwg – Grundrisse der IAI werden in DDS eingelesen (siehe Abbildung 7.1 bis Abbildung 7.4). Das Gebäudemodell wird als IFC – Datei im Format 2x2 exportiert. Abbildung 7.22 zeigt das IFC – Modell im IfcStoreyView. Das Modell enthält auf vier Stockwerken 20 Räume, die 25 Türen und 30 Fenster besitzen. Im Gegensatz zu den IAI – Modellen sind keine zusätzlichen architektonischen Elemente enthalten, z. B. Treppen-häuser.

7.5 Einsatz der IFC – Modelle in der thermischen Simulation mit RIUSKA

In diesem Teil des Praxistests werden die IAI – Modelle und das selbst erstellte DDS – Modell in RIUSKA eingelesen. Im ersten Schritt wird die visuelle Darstellung des Modells in RIUSKA auf Vollständigkeit und Zusammenhang überprüft. Weiters werden die einzelnen Räume auf ihre Umschließungsflächen und die enthaltenen Fenster und Türen überprüft. Abschließend werden die übertragenen Listen (Raumliste, Bauteilliste) in Bezug auf Raumflächen und –volumina sowie Bezeichnungen kontrolliert. Da beim Import der Modelle zum Teil erhebliche Mängel auftreten, wird der Praxistest mit einer neueren Programmversion von RIUSKA wiederholt.

7.5.1 Allgemeiner Umfang des IFC – Imports in RIUSKA

RIUSKA ist in den Versionen 4.0.13 und 4.0.26 kompatibel zu IFC2x2 und älteren Formaten. Neben der Gebäudegeometrie werden aus dem IFC – Modell Raumdefinitionen sowie Bauteilbezeichnungen übertragen. Diese Bezeichnungen können nicht geändert werden. Thermische Zonen und Raumzuweisungen zu Zonen der RLT – Anlagen werden mit der aktuellen Schnittstelle ebensowenig übertragen wie Wandaufbauten. PSETs sind in RIUSKA nicht abrufbar.

Die IFC teilen ein Gebäude in mehrere Geschoße auf. Durchgehende Schächte oder Treppenhäuser werden in der verwendeten Version nicht als solche erkannt. In den importierten IFC – Dateien müssen Räume bereits definiert sein. Betreffende Stockwerke können andernfalls nicht importiert werden.

RIUSKA importiert keine Dächer und durchgehende Geschoßdecken, da Probleme in der Raumzuweisung auftreten können (vgl. Kapitel 5.4.1). Jedem Raum wird individuell eine Decke oder ein Dach aus der Standardartikeldatenbank zugewiesen. Dachfenster können manuell hinzugefügt werden.

Die Geometrie des Modells kann in RIUSKA nicht geändert werden. Einzelne Räume und Bauteile können nachträglich gelöscht werden. Die Anordnung der Räume sowie die Lage von Wandeinbauten können nicht geändert werden. Die Parapethöhe von Fenstern bzw. die Brüstungshöhe von Türen können weder ausgelesen noch verändert werden.

7.5.2 Import der IFC – Modelle

Die Qualität des Imports der IFC – Modelle in RIUSKA wird nach folgenden Kriterien beurteilt:

- Anzahl der importierten Geschoße
- Zusammenhang der Grundrisse
- Korrekte Raumhöhen
- Anzahl der Räume
- Vorhandensein aller Umschließungsflächen der Räume
- Richtige Interpretation der Wandeigenschaften (extern, intern)
- Anzahl der importierten Fenster und Türen
- Inhalt der Raumliste (Raumbezeichnung, Fläche, Volumen)
- Durchführbarkeit einer Energiesimulation
- Durchführbarkeit einer Behaglichkeitssimulation

Gebäudemodelle werden in RIUSKA grundsätzlich ohne Dach und Geschoßdecken angezeigt. Wände sind durch einen deutlichen Abstand voneinander getrennt, um die optische Zuordnung zu einzelnen Räumen zu verdeutlichen. Außenwände werden grün, Innenwände braun dargestellt. Fenster und Türen sind blau eingefärbt. Diese Darstellung dient zur Orientierung. Änderungen können nicht vorgenommen werden.

In den nachfolgenden Abbildungen 7.23 bis 7.28 sind Screenshots der geometrischen Darstellung aller Modelle in RIUSKA 4.0.13 zu sehen. Neben den Abbildungen werden die oben genannten Kriterien die Geometrie betreffend besprochen.

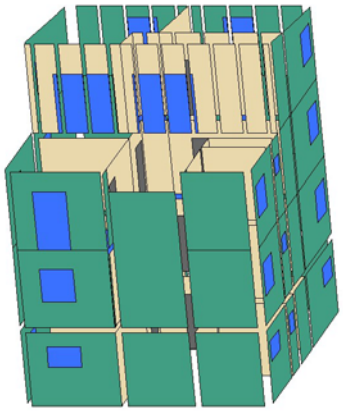


Abbildung 7.23: ADT – Modell in RIUSKA 4.0.13

Autodesk ADT

- Alle Räume sind in der korrekten Anordnung des Grundrisses enthalten.
- Die Räume sind vollständig, d. h. alle Umschließungsflächen sind vorhanden.
- Die Dachterrasse wird als eigener Raum ohne Wände geführt.
- Die Außenwand des Flurs im Obergeschoß fehlt. Die betreffende Wand des Erdgeschoßes wird nach oben verlängert.
- Im Dachgeschoß werden Wände und Fenster „zerstückelt", d. h. weiter unterteilt als für die Raumaufteilung notwendig.
- 20 Fenster und 3 Türen sind zusätzlich enthalten.
- Außenwände zur Dachterrasse hin werden als Innenwände dargestellt.

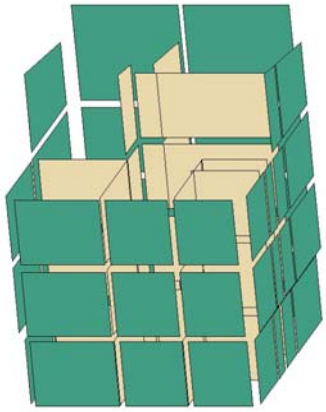


Abbildung 7.24: ArchiCAD – Modell in RIUSKA 4.0.13

Graphisoft ArchiCAD

- Alle Räume sind in der richtigen Anordnung im Grundriss enthalten.
- Die Dachterrasse besitzt keine Wände.
- An die Dachterrasse grenzende Wände fehlen auch in Nachbarräumen.
- Es sind keine Fenster und Türen vorhanden.
- An die Dachterrasse grenzende Außenwände werden als Innenwände interpretiert.

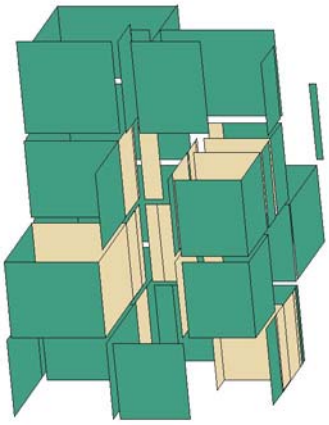


Abbildung 7.25: ALLPLAN – Modell in RIUSKA 4.0.13

Nemetschek ALLPLAN

- Alle Räume sind vorhanden. Die Anordnung stimmt in keinem Geschoß mit den ursprünglichen Grundrissen des IFC – Modells überein.
- In RIUSKA sind keine Fenster und Türen enthalten.
- Mehrere Räume sind unvollständig, d. h. Umschließungsflächen fehlen.

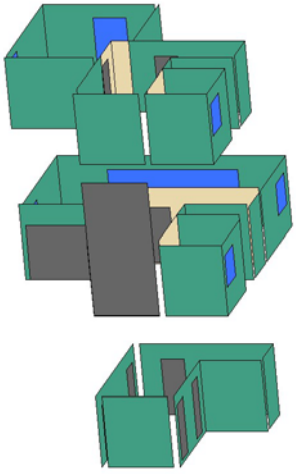


Abbildung 7.26: Bentley – Modell in RIUSKA 4.0.13

Bentley Architecture

- Alle Räume sind vorhanden. Die Anordnung stimmt in keinem Geschoß mit den ursprünglichen Grundrissen des IFC – Modells überein.
- Räume sind in RIUSKA entweder vollständig oder enthalten keinerlei Umschließungswände.
- Fenster und Türen werden nur teilweise übertragen.

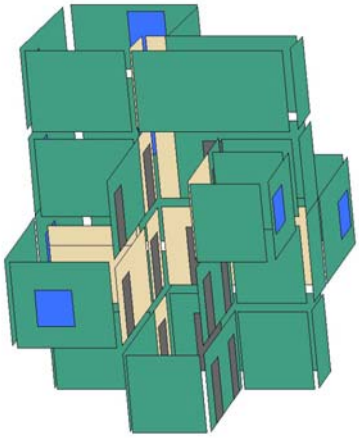


Abbildung 7.27: Revit – Modell in RIUSKA 4.0.13

Autodesk Revit Building

- Alle Räume sind vorhanden. Die Anordnung stimmt in keinem Geschoß mit den ursprünglichen Grundrissen des IFC – Modells überein.
- Räume sind teilweise vollständig, teilweise fehlen mehrere Wände.
- Fenster und Türen werden nur teilweise übertragen.

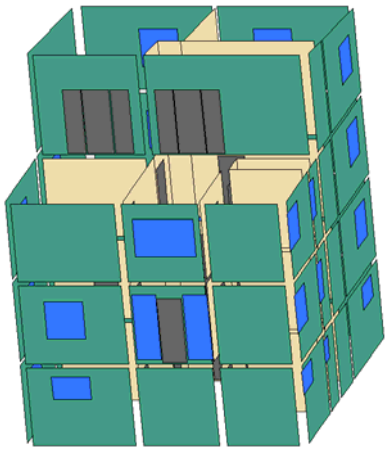


Abbildung 7.28: DDS – Modell in RIUSKA 4.0.13

Data Design System (DDS)

- Alle Räume sind vorhanden.
- Grundrisse und Raumhöhen aller Geschoße stimmen mit dem Modell überein.
- Es sind alle Fenster und Türen vorhanden.
- Alle Umschließungsflächen sind vollständig vorhanden.

Die Raumlisten, d. h. Räume, Raumbezeichnungen, Raumflächen und –volumina werden aus allen IFC – Modellen vollständig und richtig übernommen. Die Raumhöhen stimmen ebenfalls in allen Modellen mit den ursprünglichen IFC – Dateien überein.

Bei den Modellen von ALLPLAN, Bentley und Revit stimmt die Grundrissdarstellung nicht mit dem dreidimensionalen Modell überein. Abbildung 7.29 zeigt den Erdgeschoßgrundriss des ALLPLAN – Modells, in dem sich Räume teilweise überschneiden. In den Modellen von Bentley und Revit sind die Räume im Grundriss weit voneinander entfernt. Der Erdgeschoßgrundriss des Bentley – Modells ist in Abbildung 7.30 dargestellt. Die verschobenen Grundrisse haben auf die Simulation keine Auswirkung, da Räume unabhängig voneinander simuliert werden. Der Wärmefluss zwischen den Wänden wird nicht berücksichtigt.

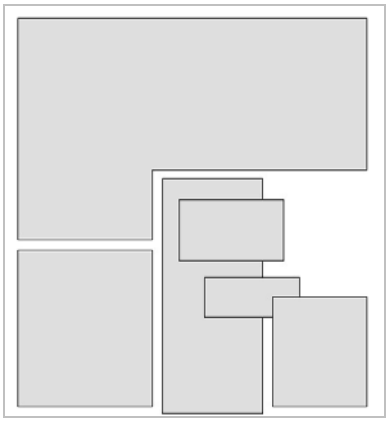


Abbildung 7.29: Grundriss EG (ALLPLAN)

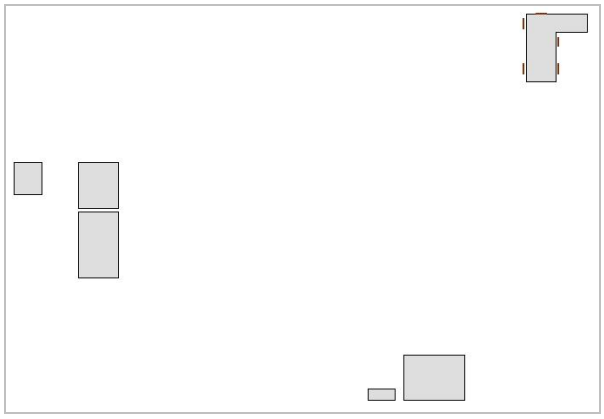


Abbildung 7.30: Grundriss EG (Bentley)

In den Modellen ADT, ArchiCAD und DDS sind Energie- und Behaglichkeitssimulationen fehlerfrei möglich. Das ALLPLAN – Modell kann trotz fehlender Wände simuliert werden. Es treten weder bei der Energie- noch der Behaglichkeitssimulation Warnmeldungen auf. Für die Modelle von Bentley und Revit ist eine Energiesimulation nicht möglich. Eine Fehlermeldung weist auf fehlende Wände hin. Eine Behaglichkeitssimulation ist nur in jenen Räumen der Modelle möglich, die mindestens eine Umschließungsfläche besitzen. Bei einer Behaglichkeitssimulation unvollständiger Räume wird keine Warnmeldung ausgegeben.

Eine Zusammenfassung der Ergebnisse des Imports in RIUSKA ist in Tabelle 7.2 dargestellt.

Tabelle 7.2: Ergebnisse des Imports der IFC – Modelle in RIUSKA 4.0.13

Kriterium	ADT	ArchiCAD	ALLPLAN	Bentley	Revit	DDS
Grundriss ok	Ja	Ja	Nein	Nein	Nein	Ja
Raumhöhe ok	Ja	Ja	Ja	Ja	Ja	Ja
Raumliste ok	Ja	Ja	Ja	Ja	Ja	Ja
Räume vollständig	Ja	Ja	Nein	Nein	Nein	Ja
Anzahl Fenster Original/RIUSKA	28 / 48	30 / 0	27 / 0	25 / 7	24 / 10	30 / 30
Anzahl Türen Original/RIUSKA	16 / 19	22 / 0	17 / 0	18 / 16	22 / 17	25 / 25
Energiesimulation	Ja	Ja	Ja	Nein	Nein	Ja
Behaglichkeitssim.	Ja	Ja	Ja	Teilweise	Teilweise	Ja

RIUSKA liest Modelle verschiedener Hersteller in variierender Qualität ein, da sich die IFC – Modelle geringfügig im Aufbau unterscheiden. Auf Anfrage wurde von Granlund ein RIUSKA – Update mit verbesserter IFC – Schnittstelle zur Verfügung gestellt. Die Ergebnisse bei einem erneuten Import der ursprünglichen IFC – Modelle zeigen zum Teil erhebliche Verbesserungen. Abbildung 7.31 bis Abbildung 7.36 stellen alle verwendeten IFC – Gebäudemodelle in RIUSKA 4.0.26 dar. Eine Übersicht der Ergebnisse des Imports ist in Tabelle 7.3 zusammengestellt.

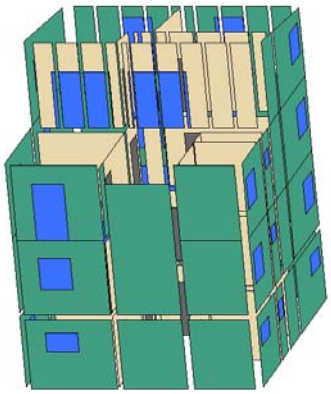


Abbildung 7.31: ADT – Modell in RIUSKA 4.0.26

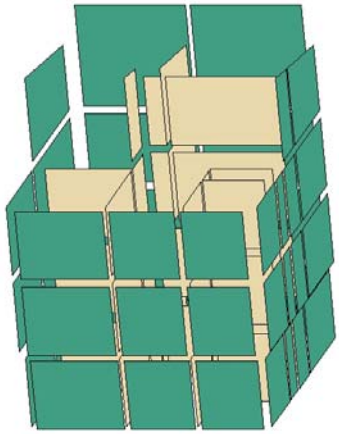


Abbildung 7.32: ArchiCAD – Modell in RIUSKA 4.0.26

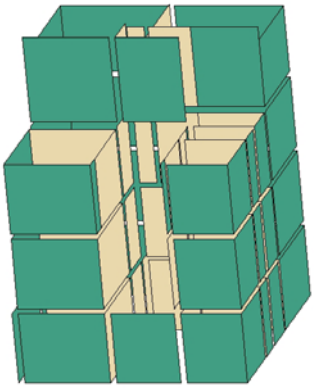


Abbildung 7.33: ALLPLAN – Modell in RIUSKA 4.0.26

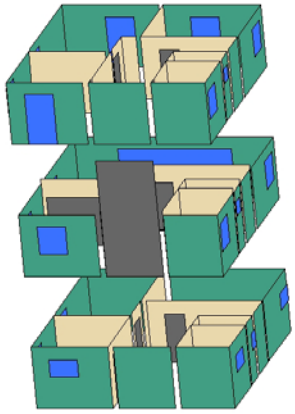


Abbildung 7.34: Bentley – Modell in RIUSKA 4.0.26

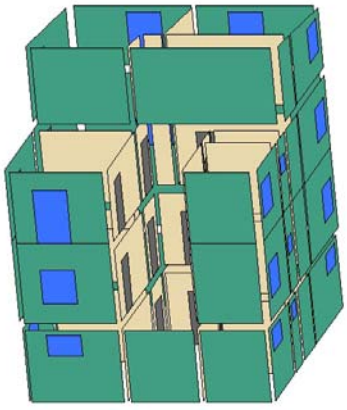


Abbildung 7.35: Revit – Modell in RIUSKA 4.0.26

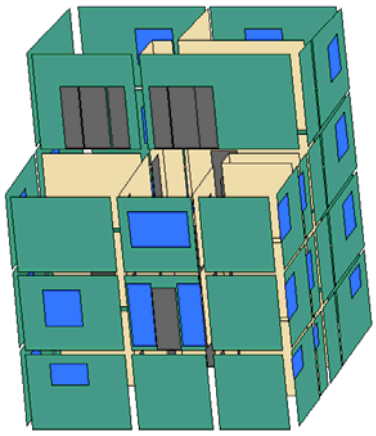


Abbildung 7.36: DDS – Modell in RIUSKA 4.0.26

Tabelle 7.3: Ergebnisse des Imports der IFC – Modelle in RIUSKA 4.0.26

Kriterium	ADT	ArchiCAD	ALLPLAN	Bentley	Revit	DDS
Grundriss ok	Ja	Ja	Ja	Ja	Ja	Ja
Raumhöhe ok	Ja	Ja	Ja	Ja	Ja	Ja
Raumliste ok	Ja	Ja	Ja	Ja	Ja	Ja
Räume vollständig	Ja	Ja	Nein	Nein	Nein	Ja
Anzahl Fenster Original/RIUSKA	28 / 48	30 / 0	27 / 0	25 / 21	24 / 24	30 / 30
Anzahl Türen Original/RIUSKA	16 / 19	22 / 0	17 / 0	18 / 17	22 / 17	25 / 25
Energiesimulation	Ja	Ja	Nein	Ja	Ja	Ja
Behaglichkeitssim.	Ja	Ja	Teilweise	Ja	Ja	Ja

Die Übertragung des ADT – Modells ist unverändert. Eine Energiesimulation für das gesamte Gebäude kann durchgeführt werden. Die Anzahl der importierten Fenster und Räume ändert sich ebensowenig wie die Übertragung der Raumliste.

In der Übertragung des ArchiCAD – Modells zeigen sich zwischen den RIUSKA – Versionen keine Änderungen. Das Modell wird vollständig importiert. Fenster und Türen fehlen auch in Version 4.0.26 in allen Räumen.

Der Import des ALLPLAN – Modells zeigt Verbesserungen. Die Grundrisse aller Stockwerke werden richtig importiert. Fenster und Türen werden auch in diesem Test nicht übertragen. Eine Energiesimulation ist nicht möglich. Behaglichkeits-simulationen sind nur in Räumen mit mindestens einer Umschließungsfläche durchführbar.

Im Bentley – Modell verbessert sich die Anzahl der importierten Fenster und Türen. Die Grundrisse aller Stockwerke stimmen in diesem Fall mit dem Modell überein. Räume im Dachgeschoß besitzen allerdings keine Wände. Im 3D – Modell werden die Geschoße voneinander abgesetzt dargestellt (vgl. Abbildung 7.34). Energie- und Behaglichkeitssimulationen sind fehlerfrei möglich.

Auch der Import des Revit – Modells zeigt Verbesserungen. Es werden alle Fenster übernommen. Die Anzahl der importierten Türen bleibt unverändert. Die Grundrisse aller Stockwerke werden richtig übertragen. Trotz teilweise fehlender Umschließungs-flächen sind alle Simulationsarten möglich.

Das DDS – Modell wird in der Update – Version von RIUSKA fehlerfrei importiert. Die Grundrisse aller Stockwerke werden ebenso übertragen wie alle Fenster und Türen. Energie- und Behaglichkeitssimulation sind möglich.

Die IFC – Modelle des IAI – Anwenderhandbuchs wurden von CAD – Anwendungsprogrammen für Architektur erstellt. Sie enthalten Zusatzinformationen, die sowohl für die Haustechnik als auch die thermische Gebäudesimulation nicht benötigt werden. Hier seien Gerüste oder Inneneinrichtungen genannt. Um auszuschließen, dass diese Zusatzinformationen zu einer fehlerhaften Interpretation des Modells in RIUSKA führen, wurden vereinfachte IFC – Dateien des gleichen Modells bei den Herstellern angefragt. Graphisoft stellte ein vereinfachtes IFC – Modell zur Verfügung dessen Gebäudemodell dem Beispiel des IAI – Handbuchs entspricht. Das Modell enthält keine Geometriedaten von Objekten, Fenstern und Türen (Schwaiger, 2006). Die Dateigröße verringert sich von 1670 [kB] auf 666 [kB]. In der Anzahl der enthaltenen Geschoße, Fenster, Türen und der Raumliste unterscheidet sich das vereinfachte Modell nicht vom ursprünglichen ArchiCAD – Modell.

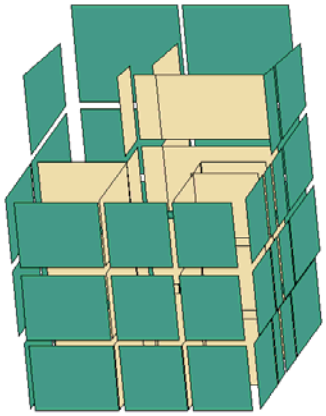


Abbildung 7.37: Vereinfachtes ArchiCAD – Modell in RIUSKA

Abbildung 7.37 zeigt das Gebäudemodell der vereinfachten ArchiCAD – IFC in RIUSKA 4.0.26. Bei der Übertragung ergeben sich keine Abweichungen zu Version 4.0.13 oder dem Import des ursprünglichen, umfangreichen ArchiCAD – Modells. Grundrisse werden korrekt übertragen. Fenster und Türen fehlen auch in der vereinfachten Version vollständig.

7.5.3 Informationsübertragung in das IFC – Modell

Ergebnisse der Behaglichkeitssimulation können in DDS übertragen und zur Anlagendimensionierung weiterverwendet werden. Die Daten werden in RIUSKA über die Exportfunktion in die IFC – Datei überspielt. Es gibt keine verstellbaren Exportoptionen.

Der in Abbildung 7.2 (Seite 71) schraffierte Raum wird für alle Modelle in RIUSKA 4.0.26 simuliert. Die Umschließungsflächen des Raumes sind in allen Modellen vollständig enthalten. Eventuell fehlende Fenster und Türen werden nicht ergänzt. Es wird eine Behaglichkeitssimulation durchgeführt, der Default – Werte aus der Artikeldatenbank zu Grunde liegen. Die Default – Werte werden unverändert für Raumprofil, Belegung, innere Lasten, Bauteilaufbauten und Standort übernommen.

In diesem Test ist das Ergebnis der Simulation nicht aussagekräftig. Ziel ist es, die Übertragung der RIUSKA – Simulationsergebnisse in DDS zu beurteilen. Der Inhalt der Raumdatenbank für den simulierten Raum wird vor und nach dem Export der Berechnungsergebnisse in die IFC – Datei aufgenommen und mit den tatsächlichen Simulationsergebnissen verglichen.

Aktiv für die weitere Berechnung wird der maximale Luftvolumenstrom aus dem IFC – Modell übernommen. Der berechnete Luftvolumenstrom ist im Raumdatendialog des Klimamoduls von DDS auslesbar. Tabelle 7.4 zeigt eine Übersicht der Simulationsergebnisse sowie des Inhalts der DDS – Raumdaten dieses Raum vor und nach dem Datenexport.

Tabelle 7.4: Ergebnisse der Informationsübertragung in die IFC – Modelle

Kennwert		ADT	ArchiCAD	ALLPLAN	Bentley	Revit	DDS
Raumdaten in DDS vor der Übertragung	Luftwechsel [1/h]	0,7	0,7	0,7	0,7	0,7	0,7
	Luftstrom [m³/h]	**0**	**75**	**77**	**66**	**83**	**69**
	max. Luftwechsel [1/h]	1,0	1,0	1,0	1,0	1,0	1,0
Simulations-ergebnisse in RIUSKA	**Luftstrom [m³/h]**	**878**	**417**	**417**	**888**	**647**	**803**
	Luftwechsel [1/h]	7,95	3,91	3,78	9,37	5,46	8,14
Raumdaten in DDS nach der Übertragung	Luftwechsel [1/h]	4512	3,91	3,78	9,38	5,46	8,14
	Luftstrom [m³/h]	**878**	**417**	**418**	**888**	**646**	**803**
	max. Luftwechsel [1/h]	222451	3,9	3,8	9,4	5,5	8,1

Beim ersten Import hinterlegt DDS Standardluftwechselzahlen und berechnet eine Vorgabe für den Luftvolumenstrom abhängig vom Raumvolumen. Der in RIUSKA ermittelte Luftvolumenstrom wird in allen Modellen korrekt übertragen. DDS ermittelt zudem eine Luftwechselrate, die nicht direkt aus den IFC importiert wird. In Tabelle 7.4 wurde die entsprechende Luftwechselzahl dem RIUSKA – Ergebnis zu Vergleichszwecken hinzugefügt. Nur für das ADT – Modell weicht die DDS – berechnete Luftwechselzahl stark von den Simulationsergebnissen ab.

In PSETs sind Zusatzinformationen zum Raum enthalten, die als Tabelle abrufbar aber nicht für Berechnungen verwendbar sind. Tabelle 7.5 zeigt die PSETs des simulierten Raumes aus dem ArchiCAD – Modell. Anzahl und Umfang der PSETs hängen vom ursprünglichen IFC – Modell ab. RIUSKA fügt den IFC – Modellen den Abschnitt *PSet_SpaceThermalDesign* hinzu. Als erstellende Anwendung wird die Vermittlungssoftware BSPro genannt. In einigen Modellen wird das von RIUSKA generierte PSET auch als *Pset_SpaceHvacInformation* bezeichnet.

Tabelle 7.5: PSET eines Raums im DDS-Raumdatendialog

IfcSpace

Name	Wert
GlobalId	24eec0dXqHseOb00sJH0zG
Name	001
LongName	Wohnen
CompositionType	.ELEMENT.
InteriorOrExteriorSpace	.INTERNAL.

PSet_Draughting

Name	Wert	Beschreibung
Layername	Räume	.
Red	0	.
Green	0	.
Blue	0	.

Pset_SpaceThermalDesign

Name	Wert	Beschreibung
VentilationAirFlowrate	0.1158 m³/s	Ventilation outside air requirement for the space.
ExhaustAirFlowrate	0.1158 m³/s	Design exhaust air flow rate for the space.
CoolingDryBulb	26 °C	Inside dry bulb temperature for cooling design
TotalHeatGain	0 W	The total amount of heat or energy gained by the space at the time of the space's peak cooling conditions.
HeatingDryBulb	21 °C	Inside dry bulb temperature for heating design
TotalHeatLoss	0 W	The total amount of heat or energy lost by the space at the time of the space's peak heating conditions.
ApplicationFullName	BSPro (Olof Granlund Oy)	.
User_FamilyName	NoPerson	.
User_Organization	The Organization	.
CreationDate	Thu May 25 15:46:17 2006	

Owner Info

Name	Wert
ApplicationFullName	ArchiCAD 9.0
ApplicationDeveloper	Graphisoft
User_FamilyName	Undefined
User_Organization	OrganizationName
CreationDate	Fri Mar 31 12:02:55 2006

7.6 Schlussfolgerung

Der Datenaustausch mittels IFC – Modellen stellt Planern Arbeitsgrundlagen auf höchstem Niveau zur Verfügung. Nur qualitativ hochwertige IFC – Modelle vereinfachen den Arbeitsablauf. Der Erstellung des Modells durch den Architekten oder Gebäudeplaner kommt daher besondere Bedeutung zu. Unabhängige Analyseprogramme sind für den nachfolgenden Planer ein wichtiges Werkzeug, um den Inhalt des Modells vorab prüfen zu können. Der Praxistest hat gezeigt, dass eine rein optische Überprüfung im IfcStoreyView keine endgültige Aussage über das Modell zulässt. So trat die im IfcStoreyView fehlerhafte Darstellung des Bentley – Modells im weiteren Verlauf des Tests nicht mehr auf.

Der Import der IAI – Modelle in die TGA – Planungssoftware DDS erwies sich als weitgehend fehlerfrei. Importiert wurden Gebäude- und Bauteilgeometrien, Raum- und Geschoßbezeichnungen sowie Bauteileigenschaften in PSETs. Das ArchiCAD – Modell zeigte die besten Übertragungseigenschaften. Mit Ausnahme des ADT – Modells können alle Modelle für die haustechnische Planung weiterverwendet werden. Beim Import des ADT – Modells fehlte ein Geschoß. Wandeigenschaften wurden ebenfalls fehlerhaft übertragen. Die photorealistische Darstellung mittels Rendering zeigte bei allen Modellen variierende Ergebnisse. Fehlende Geschoße oder Bauteile und die Farbzuweisung änderten sich wiederholt nach einem Programmneustart oder einem erneuten Import unter gleichen Importbedingungen.

Bei der Übertragung der IFC – Modelle in RIUSKA wurden Raum- und Geschoßgeometrien sowie Wandeinbauten wie Fenster und Türen importiert. Abweichungen traten in der Übernahme der Grundrisse, der Zuweisung von Umschließungsflächen zu Räumen sowie bei der Übertragung von Fenstern und Türen auf. Die Raumliste (Bezeichnung, Fläche, Volumen) wurde in allen Modellen fehlerfrei übernommen. Die Modelle von DDS und ADT zeigten die beste Übertragungsqualität. Sie können für die Gebäudesimulation verwendet werden. Die übrigen Modelle wurden teilweise unvollständig übertragen. In einigen Fällen ist eine Simulation möglich, die ob der fehlenden Bauelemente allerdings für die praktische Anwendung nicht aussagekräftig ist. Die Informationsübertragung von RIUSKA in die IFC – Datei funktionierte bei allen Modellen. Simulationsergebnisse werden mittels PSET übertragen.

Die teilweise fehlerhafte Übertragung von korrekten IFC – Modellen wies auf Probleme der IFC – Schnittstelle von RIUSKA hin. Das Entwicklungsteam von Granlund stellte auf Anfrage ein Update des Simulationsprogramms zur Verfügung. Granlund weist darauf hin, dass die Generierung der IFC – Modelle herstellerabhängig ist. Die Übernahme von IFC – Modellen in RIUSKA wird laufend verbessert und angepasst (Karola, 2006).

In diesem Stadium der Entwicklung von IFC – Schnittstellen ist der Einsatz der aktuellsten Programmversion ausschlaggebend. Im RIUSKA Update zeigten alle Modelle verbesserte oder gleichbleibende Übertragungseigenschaften. Mit Ausnahme des ALLPLAN – Modells können in allen Modellen Simulationen durchgeführt werden. Der Import von Fenster und Türen für die Modelle von ArchiCAD und ALLPLAN ist

auch im Update nicht möglich. Sinnvolle Simulationsmöglichkeiten ermöglichen DDS (beste Qualität) und ADT.

Im Praxistest erreichten vereinfachte IFC – Modelle eine ebenso hohe Übertragungsqualität in RIUSKA wie vollständige Modelle. Geometriedaten oder architektonische Zusatzinformationen können aus dem Modell entfernt werden ohne die Übertragung zu verändern. Besonders umfangreiche Gebäudemodelle erzeugen IFC – Modelle mit erheblichen Dateigrößen. Für RIUSKA können daher stark vereinfachte Modelle verwendet werden, um den Datentransfer zu erleichtern.

Im Verlauf des gesamten Praxistests zeigte sich, dass RIUSKA nur in Kombination mit einem IFC – kompatiblen CAD – Programm sinnvoll einsetzbar ist. IFC – Modelle können in RIUSKA nicht verändert werden. Um ein Modell zu testen oder anzupassen ist eine CAD – Anwendung notwendig. Der Einsatz von Analyseprogrammen und Viewern ist für die Fehlersuche empfohlen.

8 Zusammenfassung

Grundlage von thermischen Gebäudesimulationen sind zahlreiche Eingabedaten, die aus einer Vielzahl an Quellen stammen. Eine IFC – Schnittstelle ermöglicht die Übertragung virtueller Gebäude. Ihre Verwendung kann den Zeit- und Ressourcenaufwand der Datenerfassung stark verringern. Der praktische Einsatz von IFC – Gebäudemodellen wurde in dieser Arbeit an Hand des Simulationsprogramms RIUSKA und der Haustechnikanwendung DDS untersucht.

Eine Vielzahl an Projektbeteiligten bringen verschiedenste EDV – Programme in die Planung ein. Inkompatible Anwendungen machen einen Datenaustausch unmöglich. Einmal erstellte Informationen müssen manuell übertragen werden. Dieser Zeit- und Ressourcenaufwand kann durch den Einsatz von interoperabler Software vermindert werden. Als Quasi – Standard für den Austausch von Zeichnungen hat sich das Format DXF etabliert, das jedoch nur Zeichnungsinformationen enthält. Innerhalb der Bauindustrie gibt es Bestrebungen den Datenaustausch durch Standardisierung zu vereinfachen. Die VDI – Richtlinie 3805 enthält ein Modell zur Vereinheitlichung von TGA – Produktdaten, die von Herstellern zur Verfügung gestellt werden. Allgemeine Anforderungen an den Datenaustausch zwischen CAD – Systemen definiert die VDI – Richtlinie 6027. Im internationalen Standard ISO 10303 (STEP: Standards for the Exchange of Product Model Data) ist der Datenaustausch mittels Produktdaten-modellen definiert. Produktdatenmodelle basieren auf einer ganzheitlichen Betrachtung des Produkts und enthalten alle das Produkt beschreibende Daten. Grundlage der Produktdatenmodelle ist eine objektorientierte Softwareentwicklung, die auf einer sinnvollen Nachbildung real existierender Objekte beruht.

Der Bauindustrie ermöglichen Produktdatenmodelle den Einsatz von integrierten Gebäudemodellen. Das virtuelle Gebäude besteht aus einzelnen Objekten, die von Softwareprogrammen als solche erkannt werden. Ein Fenster wird als „intelligentes" Objekt dargestellt, das neben geometrischen Eigenschaften auch weitere Angaben (z. B. Material, Hersteller) enthält. Im Produktdatenmodell eines Gebäudes stellt die Geometrie nur mehr einen Teil des Gesamtmodells dar. Der Mehrwert eines Produktdatenmodells besteht in der Verknüpfung von geometrischen und topologischen Daten mit semantischen Eigenschaften des Gebäudes. Semantische Eigenschaften sind unabhängig von der Ausdehnung oder Lage eines Objekts und beziehen sich u. a. auf Materialwerte, Nutzungsrechte und andere Objekteigenschaften. Abbildung 8.1 zeigt den Inhalt eines vereinfachten Gebäudemodells. In Analogie zu einem reinen Geometriemodell bezieht sich der dargestellte Raum auf Ursprungspunkte und dessen Koordinaten. Darüber hinaus sind Wände als eigenständige Objekte enthalten. Sie besitzen sowohl geometrische (Länge, Höhe, Tiefe) als auch nicht – geometrische Eigenschaften (Bautyp, angrenzende Räume). Das Modell enthält weiters abstrakte Objekte in Form von Räumen, Funktionen etc. Alle Objekte stehen in Beziehung zueinander.

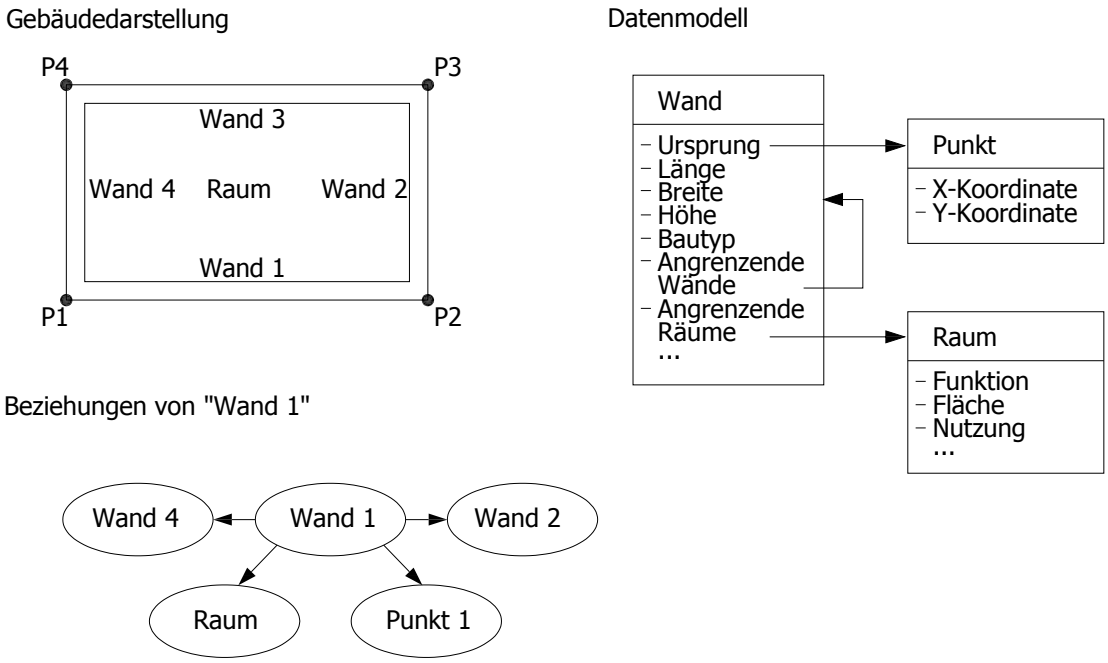


Abbildung 8.1:Gebäudedatenmodell (nach Khemlani, 2004)

Das Produktdatenmodell deckt den gesamten Lebenszyklus des Gebäudes ab. Es wird vom Architekten erstellt, während der Bauphase von anderen Planern erweitert und steht in der Betriebsphase des Gebäudes dem Facility Management zur Verfügung. Der Begriff BIM (Building Information Modeling) bezeichnet die Verwendung von virtuellen Gebäudemodellen. Das Datenformat eines BIM hängt von der verwendeten Software ab. Produktdatenmodelle können dann sinnvoll eingesetzt werden, wenn sie möglichst vielen Anwendungen zur Verfügung stehen. Ein herstellerunabhängiges Produktdatenmodell, das in Form einer Schnittstelle in CAD – Applikationen implementiert wird, ermöglicht die notwendige Software-interoperabilität.

Die Industry Foundation Classes (IFC) sind ein herstellerunabhängiges, speziell für die Bauindustrie entwickeltes Produktdatenmodell für Gebäude bzw. Gebäude-komplexe. Seit 1995 entwickelt die International Alliance for Interoperability (IAI) das IFC – Produktdatenmodell. Unter dem Dach der IAI wird das Modell mit Unterstützung führender Bausoftware – Hersteller weiterentwickelt. Ziel der IAI ist es, die IFC – Schnittstelle als internationalen, herstellerunabhängigen Standard für den Austausch von virtuellen Gebäudemodellen zu etablieren.

Die Datenstruktur des IFC – Produktdatenmodells wird in vier Layer unterteilt (Abbildung 8.2). Die unteren Hierarchieebenen definieren die generische Datenstruktur des Modells. Hierzu zählen *Resource* und *Core Layer*. Die obersten Ebenen der Hierarchie sind Grundlage des Datenaustausches. Der *Interoperability Layer* ermöglicht die Übertragung grundlegender Elemente des Gebäudemodells, die von den meisten Applikationen mit IFC – Schnittstelle unterstützt werden. Wichtigstes Modul dieses Layers sind die *Shared Building Elements*, die den Geometrieaustausch ermöglichen.

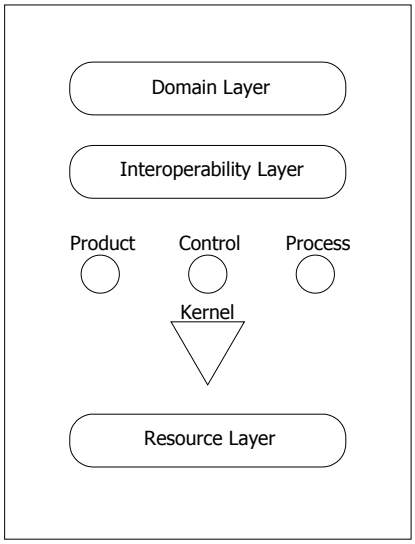


Abbildung 8.2: Struktur der IFC – Layer

Ein Datenaustausch mit höherem Detaillierungsgrad wird über die einzelnen Module des *Domain Layers* ermöglicht. Diese Form des Datenaustausches erlaubt die Übertragung von Teilmodellen einzelner Gewerke zwischen entsprechend spezialisierten Applikationen. Der Domain Layer enthält das Modul *HVAC*, das für die Übertragung haustechnischer Anlagen und Komponenten verwendet wird. Voraussetzung für die Übertragung von haustechnischen Anlagen ist die Unterstützung des HVAC – Moduls in den verwendeten IFC – Schnittstellen.

Die IFC – Version 2x wurde als ISO/PAS 16739 (IFC2x – Plattform) zertifiziert. Spätere Versionen basieren auf dieser Plattform. Die aktuellste Version der IFC – Schnittstelle wird mit 2x3 bezeichnet. Von der IAI zertifizierte Softwareanwendungen der Version 2x2 sind zurzeit auf dem Markt erhältlich. Für den Austausch nicht – geometrischer Daten steht eine XML – Version der IFC zur Verfügung (ifcXML).

Softwareanwendungen können IFC – Kompatibilität durch eine integrierte IFC – Schnittstelle erreichen. Externe Vermittlungsprogramme, die das IFC – Modell in das interne Datenformat der Anwendung übertragen, können ebenfalls verwendet werden. Der Austausch des Gebäudemodells erfolgt über einzelne Dateien. Die benötigte Speicherkapazität umfangreicher Gebäudemodelle führte zur Entwicklung von Modell Servern. Das IFC – Gebäudemodell wird auf einem zentralen Server zur Verfügung gestellt. Anwender greifen nur auf benötigte Teile des Modells zu. Diese Form der geteilten Datenverwaltung ermöglicht eine einfachere Kontrolle von Zugriffs- und Bearbeitungsrechten. Der Einsatz von Modell Servern befindet sich zurzeit in einem Anfangsstadium. Besonders Fragen des Projektmanagements (Berechtigungen, Versionierung…) bedürfen weiterer Entwicklung.

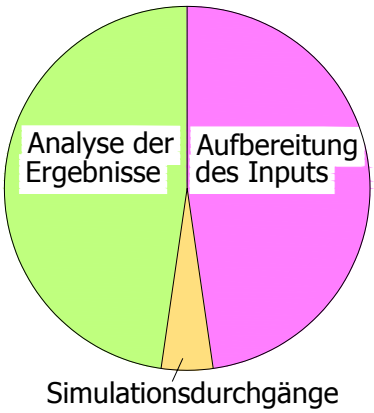


Abbildung 8.3: Verteilung der Gesamtkosten einer Energiesimulation (nach Bazjanac, 2001)

Dem Gebäudemodell kommt in der thermischen Gebäudesimulation besondere Bedeutung zu. Es ist Berechnungsgrundlage und ausschlaggebend für die Qualität der Simulation. Betrachtet man die Gesamtkosten einer thermischen Gebäudesimulation stellen die Dateneingabe und die Ergebnisauswertung den Großteil der Kosten dar (siehe Abbildung 8.3). Die eigentliche Simulation erfordert einen vergleichsweise geringen Aufwand.
Der Anteil der Dateneingabe kann wiederum in mehrere Teilbereiche unterteilt werden (siehe Abbildung 8.4). Aufbereitung und Eingabe der Gebäudegeometrie haben den größten Anteil an der Gesamtdatenaufbereitung vor der Durchführung einer Simulation. Die Fehlerbehebung ist ebenfalls ressourcenaufwändig. Vereinfachungen in der Geometrieerfassung durch Übernahme bestehender Modelle oder Modellteile wirkt sich auf den gesamten Simulationsvorgang positiv aus, da die aufwändige und fehleranfällige manuelle Eingabe minimiert wird.

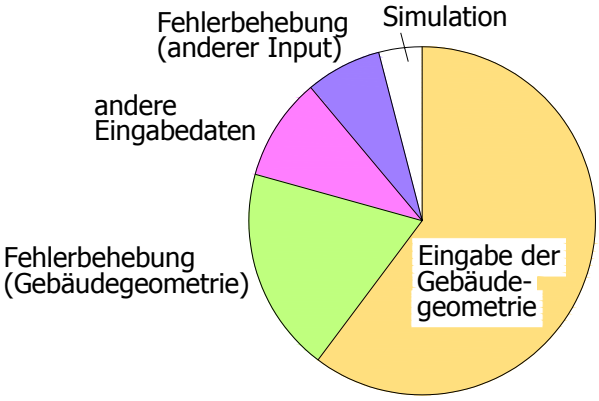


Abbildung 8.4: Anteil der Gebäudegeometrie an der Aufbereitung der Eingabedaten (nach Bazjanac, 2001)

Traditionell werden die Eingabedaten einer Gebäudesimulation aus einer Vielzahl an Quellen zusammengetragen. Die Daten werden manuell in ein unabhängiges Simulationsprogramm eingegeben. Dieses Prinzip ist trotz verbesserter Eingabeoberflächen aufwändig und fehleranfällig. Die IFC – Schnittstelle ermöglicht den direkten Datenaustausch zwischen Softwareanwendungen. Das Hauptaugenmerk IFC – kompatibler Simulationsprogramme liegt derzeit in der Übertragung der Gebäudegeometrie. In der thermischen Betrachtungsweise unterscheiden sich Gebäude stark von einer architektonischen Sichtweise. Eine einfache Wand kann in einem Simulationsmodell mehrere Teilabschnitte besitzen, die verschiedenen Zonen zugeordnet sind. Trotz des höheren Detaillierungsgrad der Zonierung enthalten Gebäudemodelle vielfach Informationen (z. B. zusätzliche Geometrien), die für Simulationen irrelevant sind. Um ein IFC – Modell an die Anforderungen des Simulationsprogramms anzupassen, werden für den IFC – Import Vermittlungsprogramme eingesetzt.

Das weitest entwickelte Simulationsprogramm mit IFC – Schnittstelle ist EnergyPlus. Es kann sowohl Geometriedaten als auch HVAC – Anlagen übernehmen. EnergyPlus verwendet die Vermittlungssoftware BSPro COM Server. Über ein externes Modul können weitere Simulationsprogramme IFC – Modelle verarbeiten. Die CAD – Anwendung SimCAD besitzt eine IFC – Schnittstelle und ermöglicht dem Simulationsprogramm TRNSYS die Geometrieübernahme aus IFC – Dateien.

Die finnische Planungs- und Beratungsfirma Olof Granlund Oy entwickelt seit 1996 das Simulationsprogramm RIUSKA. Die Anwendung wird für die dynamische Simulation des Energieverbrauchs lüftungstechnischer Anlagen und der Behaglichkeit in Gebäuden eingesetzt. RIUSKA wurde für den praktischen Einsatz in der Planungsphase und für das Facility Management optimiert. Das Programm ist eine benutzerfreundliche Windows – Oberfläche des Simulationskerns DOE-2.

RIUSKA – Simulationen basieren ausschließlich auf bestehenden IFC – Gebäudemodellen, die über eine IFC2x2 – Schnittstelle importiert werden. Das Programm ist Client der Vermittlungssoftware BSPro COM Server und erstellt selbst keine IFC – Dateien. Aus dem IFC – Modell werden Grundrisse, Umschließungsflächen, Lage und Größe von Fenstern und Türen sowie Raum- und Bauteilbezeichnungen importiert. Bauteileigenschaften (Materialien, u – Werte) werden nicht übernommen. Simulationsergebnisse (Luftvolumenströme, Auslegungstemperaturen, Zonenaufteilung) können als PSETs in das Gebäudemodell übertragen werden. PSETs enthalten Zusatzinformationen und sind in Form von Tabellen an ein Objekt (Fenster, Wand) angehängt.

Im Rahmen eines Anwenderhandbuchs stellt die IAI fünf IFC – Gebäudemodelle eines Einfamilienhauses zur Verfügung. Die Modelle wurden von den führenden Softwareanwendungen für Architektur erstellt: Autodesk Architectural Desktop (ADT), Graphisoft ArchiCAD, Nemetschek ALLPLAN, Bentley Architecture und Autodesk Revit.

Um die Verwendung der IFC – Schnittstelle in der Praxis zu dokumentieren, wurden diese fünf Beispielmodelle in RIUSKA und in die TGA – Planungssoftware von Data Design System (DDS) eingelesen. Zuvor wurden die Modelle in der unabhängigen Analysesoftware *IfcStoreyView* überprüft, die keine Fehler der Modelle feststellte.

Die Modelle konnten weitgehend fehlerfrei in DDS importiert werden. Gebäude- und Bauteilgeometrien, Raum- und Geschoßbezeichnungen sowie Bauteileigenschaften in PSETs wurden übernommen. Das ArchiCAD – Modell zeigte die besten Übertragungseigenschaften. Mit Ausnahme des ADT – Modells, bei dem ein unvollständiger Import stattfand, konnten alle Modelle für die haustechnische Planung weiterverwendet werden. Abweichungen zeigten alle Modelle in der photorealistischen Darstellung im Rendering. Die Darstellung variierte bei mehreren Importversuchen und Programmneustarten.

Bei der Übertragung der IFC – Modelle in RIUSKA (Version 4.0.13) traten Abweichungen in der Übernahme der Grundrisse und der Zuweisung von Umschließungsflächen zu Räumen auf. Einige Räume wurden ohne Wände dargestellt. Die Anzahl der importierten Fenster und Türen wich teilweise stark vom ursprünglichen IFC – Modell ab. So konnten keine Wandöffnungen der Modelle von ArchiCAD und ALLPLAN importiert werden. Beste Übertragungsqualitäten wiesen das ADT – Modell und ein selbst erstelltes DDS – Modell auf. Sie konnten für eine aussagekräftige Gebäudesimulation verwendet werden.

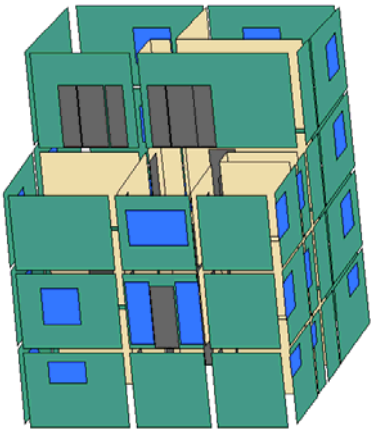


Abbildung 8.5: DDS – Modell in RIUSKA (Version 4.0.26)

Abbildung 8.5 zeigt das in DDS erstellte Modell in RIUSKA. Die Darstellung des Gebäudes ist vollständig, da Dächer und Geschoßdecken grundsätzlich nicht angezeigt werden. Wände werden versetzt dargestellt, um die Zuordnung zu einzelnen Räumen zu verdeutlichen.

Die Übertragung eines berechneten Luftvolumenstroms von RIUSKA in das IFC – Modell funktionierte für alle Modelle. Den Räumen wurde ein PSET hinzugefügt, das die Simulations-ergebnisse in Form des Maximalwerts enthält.

Da die ursprünglich fehlerfreien IFC – Modelle teilweise unvollständig in RIUSKA importiert wurden, stellte Granlund ein Programmupdate (Version 4.0.26) zur Verfügung. Die folgenden Abbildungen zeigen das Revit – Modell in der ursprünglichen Programmversion (Abbildung 8.6) und im Update (Abbildung 8.7). Alle Modelle zeigten im Update eine verbesserte oder gleich bleibende Qualität der Übertragung. Die besten Übertragungsergebnisse zeigten auch hier ADT und DDS.

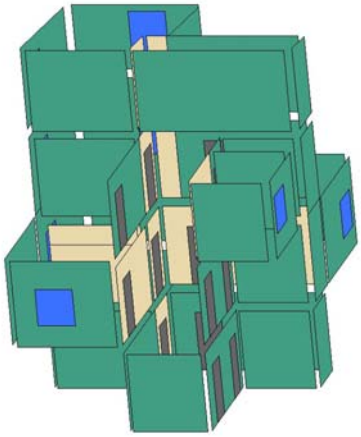


Abbildung 8.6: Revit – Modell in RIUSKA (Version 4.0.13)

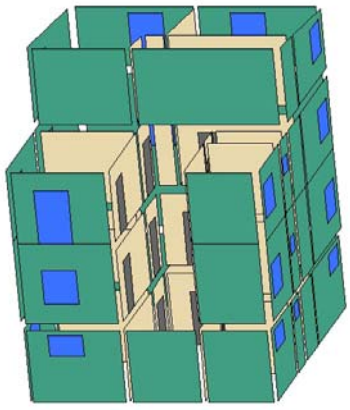


Abbildung 8.7: Revit – Modell in RIUSKA (Version 4.0.26)

Der Praxistest zeigt, dass in diesem Stadium der IFC – Entwicklung der Einsatz der aktuellsten Programmversionen von besonderer Bedeutung ist, da die IFC – Schnitt-stellen kontinuierlich weiterentwickelt werden. Die Anwendung des IFC – Produktdatenmodells beschränkt sich zurzeit auf Pilotprojekte. Viele Programmversionen liegen erst im Beta – Stadium vor. Eine Verbesserung des Modells kann durch Rückmeldungen aus dem kontinuierlichen Praxiseinsatz erreicht werden. Die Vorteile von virtuellen Gebäudemodellen, die über eine IFC – Schnittstelle übertragen werden, können in der thermischen Gebäudesimulation erst zum Tragen kommen, wenn die Erstellung einer IFC – Datei für den Architekten oder Gebäudeplaner zur Routine geworden ist und Modelle sowohl fehlerfrei erstellt als auch importiert werden können.

9 Glossar

Die *kursiv* gekennzeichneten Erläuterungen wurden von Hörenbaum (2002) übernommen.

4D – Visualisierung	Visualisierung mit einer zeitlichen Komponente
Abstrakte Klasse	Klasse, die nicht in einer Austauschdatei instanziiert werden kann
AEC	Architecture – Engineering – Consulting, englische Kurzbezeichnung von Bau- und Baunebengewerbe
Anlagensimulation	nach VDI 6020: Berechnung des stundenweisen Verhaltens einer heiz- oder raumlufttechnischen Anlage aufgrund einer vorher durchgeführten Gebäudesimulation.
Anwendungsprogramm	Software zur Lösung bestimmter Aufgaben sowie zur Erstellung von Dokumenten und Plänen, z.B. CAD- oder Statikprogramme
AP	Application Protocol: Produktmodell innerhalb von ISO 10303
AP 225	Application Protocol 225 – Building Elements Using Explicit Shape Representation: In ISO 10303-225 definiertes Produktmodell für das Bauwesen
API	Application Programming Interface: Programmier-schnittstellen von Softwareprogrammen
Applikation, Applikations-software	Anwendungsprogramm
ASCII	American Standard Code for Information Interchange
Attribut	Bestandteil eines Entities zur Beschreibung einer Eigenschaft
Auflösung	Datenstruktur, bei der ein Bauelement in mehrere Komponenten aufgelöst wird. Diese Komponenten sind entweder ebenfalls Bauelemente oder Bauteile. In den IFC wird diese Struktur durch die Klasse *IfcRelDecomposes* dargestellt.

Austauschdatei	Einfachste Form des Produktdatenaustausches, in der die Klasssen eines Produktmodells in ASCII – Dateien instanziiert werden.
Bauinformatik	Fachgebiet innerhalb des Bauingenieurwesens, das sich mit der bauingenieurspezifischen Anwendung der Computerwissenschaft befasst.
Behaglichkeit	Hier: Zustand, in dem der Mensch mit seiner thermischen Umgebung zufrieden ist. Man fühlt sich „thermisch neutral" und wünscht keine kältere oder wärmere Umgebung.
Benutzeroberfläche	Teil eines Computerprogramms der den Datenaustausch mit dem Benutzer ermöglicht (Datenein- und ausgabe).
BIM	Building Information Modeling: Bauteilorientierte Erarbeitung eines virtuellen Gebäudes. Die IFC sind ein mögliches Datenaustauschformat für BIM.
BLAST	Building Loads Analysis and System Thermo-dynamics. Simulationsprogramm zur thermischen Analyse von Gebäuden. Inputdateien werden über die Benutzeroberfläche HBLC (Heat Balance Loads Calculator) erstellt. Das BLAST Support Office befindet sich an der University of Illinois.
BLIS	Building Lifecycle Interoperable Software: von der IAI unabhängige Initiative zur Koordination der IFC Implementierung.
CAAD	Computer Aided Architectural Design: CAD – Software für Architektur
CAD	Computer Aided Draughting oder Computer Aided Design: Sammelbegriff für Software für technisches Zeichnen oder Entwerfen
CAM	Computer Aided Manifacturing: Rechnergestützte Fertigung.
CAFM	Computer Aided Facility Management
Client	Arbeitsstation in einem Netzwerk, die von einem Server angebotene Ressourcen benutzt.
Client – Server – System	Netzwerkstruktur, bei der Server zentrale Ressourcen bereitstellen, die über Arbeitsstationen (Clients) bearbeitet werden können.

Core Layer	Funktionale Schicht der IFC, die generische Strukturen in Form des Kernels und der Core Extensions beinhaltet.
Datenmodell	Hier: Formale, modellhafte Definition eines Datenformates. Das Datenmodell ist Kernstück eines Produktmodells.
DDS	Data Design System: objektorientiertes und IFC – kompatibles CAD – Programm mit Zeichnungs- und Berechnungsmodulen für Elektrotechnik und TGA.
DOE	Departement of Energy: Abteilung der US – Regierung zur Entwicklung und Förderung von Energietechnologien.
DOE-2	Simulationsprogramm ohne Benutzeroberfläche zur Gebäude- und Anlagensimulation sowie der stündlichen Berechnung von Energiekosten. DOE-2 wurde vom LBNL entwickelt. Die letzte Version ist DOE-2.1E.
Domäne	Hier: Bereich eines Produktdatenmodells, der eine bestimmte Anwendung betrifft.
Domain Layer	Funktionale Schicht der IFC, die Teilmodelle für verschiedene Anwendungen enthält (z.B. FM, HVAC).
dwg	Datenaustauschformat des Softwareherstellers AutoDesk
dxf	Drawing Exchange Format: herstellerabhängiges Zeichnungsformat des CAD Programms AutoCAD
Dynamische Eigenschafts-definition	Semantische Eigenschaft, deren Bedeutung durch Vereinbarung außerhalb des eigentlichen Datenmodells definiert wird. In den IFC wird dies durch die sogenannten PSETs erreicht.
Eigenschaftsdefinition, Eigenschaftsklasse	Klasse der IFC, die semantische Eigenschaften für ein Objekt definiert.
Einfachvererbung	Festlegung in den IFC, dass eine Unterklasse nur eine einzige unmittelbare Oberklasse besitzen darf.
EnergyPlus	Gebäudesimulationsprogramm, das BLAST und DOE-2 vereint. Wird u.a. vom LBNL entwickelt.

Entity	Nach ISO 10303 der wichtigste Datentyp in EXPRESS. Ein Entity wird definiert zur Darstellung von Objekten oder deren Zuständen. In den IFC tritt der Begriff der Klasse anstelle des Entities.
EXPRESS	Formale Spezifikationssprache nach ISO 10303-11 für Datenmodelle
EXPRESS – G	Graphische Variante von EXPRESS
EXPRESS – Schema	Satz von Regeln und Datendefinitionen in der Sprache EXPRESS
EXPRESS – X	Formale Sprache nach ISO 103030-14 zur Definition des Mappings von Daten aus einem EXPRESS Datenmodell in ein anderes.
FM	Facility Management: Bewirtschaftung von Gebäuden
Gebäudesimulation	nach VDI 6020: Stundenweise Berechnung der Raumreaktion (Last, Temperatur) unter Berücksichtigung aller inneren und äußeren Einflüsse. Für die Gebäudesimulation wird der Wetterdatensatz eines Jahres benötigt.
Gebäudetopologie	Topologische Strukturen im Objektmodell eines Gebäudes, durch die z. B. Fenster in Wänden verankert sind, oder mit denen sich Räume über ihre angrenzenden Wände und Decken definieren.
Generische Datenstruktur	Struktur innerhalb eines Datenmodells, die selbst zu allgemein ist, um angewandt zu werden.
Geometrische Modellierung	Datenmodellierung, die lediglich geometrische Daten beinhaltet.
HVAC	Heating, Ventilation, Air Conditioning; englische Kurzbezeichnung der wichtigsten Gewerke der Haustechnik.
IAI	Internationale Allianz für Interoperabilität: Nicht kommerzielle Organisation mit dem Ziel, den Datenaustausch im gesamten Bauwesen zu ermöglichen
IBPSA	International Building Performance Simulation Association: Nicht kommerzielle Organisation von Forschern, Entwicklern und Anwendern der thermischen Gebäude- und Anlagensimulation
IFC	Industry Foundation Classes: Produktmodell der IAI

ifcXML	XML – Version der IFC, das für den Austausch von semantischen Teilbereichen eines IFC Gebäudemodells geeignet ist (z.B. Produktdaten-banken).
Implementierung	Hier: Einbindung eines Produktmodells in Applikationssoftware
Implementation Level	Nach ISO 10303: Art der Implementierung. […]
Implementierungsbereich	Sicht
Integrale Planung	Vision der ganzheitlichen Planung von Bauwerken, die deren gesamten Lebenszyklus, alle am Bau Beteiligten sowie die Nutzer berücksichtigt.
Interoperabilität	Produktdatenaustausch
Instanz	Nach ISO 10303: Ein mit konkreten Werten belegtes Entity in einer Austauschdatei, einem Repository oder einer Datenbank.
Instanziierung	Vorgang des Belegens eines Datenmodells mit konkreten Instanzen.
Interoperability Layer	Funktionale Schicht der IFC, die jene Entities beinhaltet die von den meisten Anwendungen ausgetauscht werden (z.B. Shared Building Elements).
ISO	International Organization for Standardization
Kernel	Hier: Kern des IFC Produktdatenmodells im Core Layer, der die Prozess- und Datenorganisation festlegt.
Klasse	In Programmiersprachen ist eine Klasse eine Einheit, die sowohl über Eigenschaften als auch über Methoden verfügt. […]
Konkrete Klasse	Im Gegensatz zur abstrakten Klasse kann eine konkrete Klasse in einer Austauschdatei instanziiert werden.
Konformität	Nach ISO 10303: Soll – Zustand einer Implementierung eines Produktmodells, bei dem alle syntaktischen, formalen und inhaltlichen Vorgaben des Produktmodellstandards erfüllt sind.

Layer	Hier: Funktionale Schichten der IFC, die hierarchisch und modular aufgebaut sind. Das IFC Produktdatenmodell besitzt vier Layer: Resource, Core, Interoperability und Domain Layer.
LBNL	Lawrence Berkeley National Laboratory: Forschungszentrum des US Department of Energy an der University of California
Leaf Node Class (LNC)	Konkrete Klasse, die einen Ast im Vererbungsbaum eines Datenmodells beendet.
Mapping	Nach ISO 10303: Übertragen von Daten aus einem Format in ein anderes. Das Mapping von programminternen Daten in eine Austauschdatei wird auch als Instanziierung bezeichnet.
Metasprache	Eine Metasprache beschreibt andere Sprachen. Der Vorteil von Metasprachen besteht darin, dass sie den Aufwand für die Entwicklung von Parsern (Übersetzern) verringern.
Modellierungsart	Hier: Die Modellierungsart bestimmt den Inhalt und die Darstellungsart eines Datenmodells. Es wird unterschieden in geometrische, topologische, semantische, featurebasierte und objektorientierte Modellierung.
Modellkern	Innerster Teil eines Datenmodells
Oberklasse	Klasse, die ihre Eigenschaften an eine oder mehrere Unterklassen vererbt.
Objectified Relationship	Regel in den IFC, gemäß der Objekte nicht direkt aufeinander verweisen, sondern über Relationsklassen verknüpft werden.
Objekt	In Programmiersprachen ist ein Objekt eine instanziierte Klasse. […]
Objektmodell	Darstellung eines Bauwerks in einem Produktmodell oder einer objektorientierten Software anhand von bauteilbezogenen Objekten anstatt durch Zeichnungsdaten.
Objektorientierte Datenmodellierung	Modellierungsart, bei der reale Sachverhalte anstelle von EDV – orientierten Strukturen modelliert werden. Die Objekte enthalten alle Daten, die das reale Vorbild besitzt. Charakteristisch ist die Verwendung von Objektkapselung und Vererbung.

Objektorientierte Software	Anwendungssoftware, die mit einem Objektmodell arbeitet. Für den Anwender eines CAD – Programms beispielsweise äußert sich dies dadurch, dass er nicht mit Strichen arbeitet, sondern mit „intelligenten" Objekten (Wände, Decken, Stahlträger etc.)
PAS	Publicly Available Specification: Richtlinien oder Vereinbarungen von Arbeitsgruppen, die frei zugänglich sind. Die ISO kann PAS in den Normenstand übernehmen (z. B. die Normung der IFC2x Plattform in der ISO/PAS 16739).
PMV	Predicted Mean Vote: Index, der die erwartete mittlere Beurteilung eines Raumklimas durch eine große Gruppe an Menschen angibt.
PPD	Predicted Percentage of Dissatisfied: Index, der den erwarteten Anteil Unzufriedener bei einem bestimmten PMV – Index angibt.
Produktdaten	EDV – Daten, die ein Produkt als Ganzes beschreiben und nicht nur Teilaspekte.
Produktdatenaustausch	Kommunikation von Anwendungsprogrammen über ein Produktmodell, im Idealfall ohne Datenverlust.
Produktmodell	Modellhafte Darstellung eines Bauwerks zum Zweck des Produktdatenaustauschs. In dieser Arbeit ist hiermit immer der gesamte Standard gemeint, dessen Kernstück das Datenmodell ist.
Projektstruktur	Aus Projektbeschreibung, Darstellungskontext und Festlegung von Maßeinheiten bestehender, äußerster Rahmen einer IFC – Austauschdatei.
PSD	Property Set Definition markup language: XML – basierte Sprache zur Definition dynamischer Eigenschaften in den IFC.
PSET	Dynamische Eigenschaftsdefinition in den IFC durch *IfcPropertySet.*
Relation, Relationsklasse	Spezielle Klasse der IFC, mit der Beziehungen zwischen Objekten dargestellt werden.
Rendering	Erzeugung eines digitalen (3D-) Bildes aus einer Bildbeschreibung. Hier: photorealistische Darstellung eines Gebäudemodells

Repository	Software – Archiv zur Aufbewahrung von Quellen für Programme und Dokumente oder strukturierten Produktdaten.
Resource Layer	Funktionale Schicht der IFC, die allgemeingültige, grundlegende Datenstrukturen beinhaltet, die von höhergeordneten Layern genutzt werden können (z.B. Maßeinheiten, Materialdefinitionen, Geometrie).
Ressource	Hier: Teil eines Datenmodells, in dem ein grundlegendes und allgemeingültiges Darstellungskonzept definiert ist, z. B. für Maßeinheiten oder Geometrie.
RLT – Anlage	Raumlufttechnische Anlage
SABLE	Simple Access to the Building Lifecycle Exchange: Projekt der IAI zur Harmonisierung der IFC Model Server durch Definition von einfachen API für IFC Model Server und hoch spezialisierte API für einzelne Domänen der IFC.
SDAI	Standard Data Access Interface: In ISO 10303 standardisierte Programmierschnittstelle zur Implementierung von Produktmodellen
Semantik	Lehre von der Wortbedeutung. In der Datenverarbeitung werden nichtgeometrische und nichttopologische Eigenschaften als semantisch bezeichnet. Die Bedeutung dieser Eigenschaften kann nicht abgeleitet, sondern muss vereinbart werden.
Server	Zentraler Rechner in einem Netzwerk, der den Arbeitsstationen Daten, Speicher und Ressourcen zur Verfügung stellt.
Selektiver Import	Beim selektiven Import von Daten sucht ein Programm sich bestimmte Instanzen aus einem Datenbestand heraus, die es interpretiert. Die übrigen Daten werden ignoriert und gehen verloren.
Sicht	Hier: Eine Sicht definiert eine Untermenge eines Datenmodells und stellt Regeln zur Behandlung des Datenbestandes Gruppen bestimmter Anwendungsprogramme. Eine „CAD – Sicht" würde z.B. die für CAD – Programme relevanten Teile eines Modells beinhalten.

Simulation	nach VDI 6020: Simulation bezeichnet die Nachbildung tatsächlicher Vorgänge an einem Modell sowie die Nachahmung von Realprozessen mit einem Computer.
SPF	Step Physical File: Austauschdatei im STEP – Format mit der Dateinamenerweiterung *.stp oder *.ifc
SQL	Structured Query Language: Datenbanksprache
Statische Eigenschafts-definition	Semantische Eigenschaft, deren Bedeutung innerhalb der Klassendefinition eines Datenmodells vereinbart wird.
STEP	Standard for the Exchange of Product Data: Kurzname von ISO 10303
STEP – Kompatibilität	Ein Produktdatenmodell ist kompatibel zu ISO 10303 (STEP), wenn es gemäß der dort vorgeschriebenen Modellierungsmethoden definiert ist und die entsprechenden Ressourcen unterstützt.
TGA	Technische Gebäudeausrüstung
Top – Down – Ansatz	Prinzip der IFC, nach dem ein Bauwerk als Ganzes dargestellt wird und mit steigender Detaillierung in seine Bestandteile aufgelöst wird.
Topologie	Lehre von der Lage und Anordnung geometrischer Gebilde im Raum
Unterklasse	Klasse, die einen Teil ihrer Eigenschaften aus einer oder mehreren Oberklassen erbt.
Vererbung	Paradigma in der Datenmodellierung, bei dem Klassen ihre Eigenschaften an Unterklassen weitergeben, also sozusagen „vererben".
Vererbungsbaum	Darstellung einer Klassenhierarchie in Baumform
VDI	Verein Deutscher Ingenieure: erarbeitet u.a. die „VDI – Richtlinien" als anerkannte Regeln zum Stand der Technik.
Visualisierung	Fotorealistische Darstellung von räumlichen Objekten im Computer

Wetterdaten	Datensatz, der stündliche Parameter des Wetters an einem bestimmten Standort enthält. Inhalt und Format des Wetterdatensatz hängen vom Simulationsprogramm ab. Für die Erstellung des Datensatzes kommen reale Messungen, sowie statistische Berechnungen in Frage.
Wissensbasiertes System	Ein wissensbasiertes System führt Operationen mit Unterstützung von gespeicherten Fakten und sonstigem Wissen aus. Ein Beispiel aus dem Bauwesen ist eine Schadensdatenbank, aus dem der Benutzer Informationen anhand ähnlicher Fälle erhält.
XML	Extensible Markup Language: dokumentorientierte Metasprache zur Definition von Dokumentenformaten. […]
XML – Schema	XML – Sprache zur Definition von XML – Vokabularen, durch die XML von der Dokument-orientierung zur wirklichen Datenorientierung erweitert werden kann.
XSD	XML – Schema – Definition: konkretes XML – Schema.
XSLT	Programmiersprache zur Transformation von XML – Dokumenten.
Zeichenprogramm	CAD – Programm, das im Gegensatz zu objektorientierten CAD – Programmen lediglich mit geometrischen Objekten arbeitet.

10 Literaturverzeichnis

Bazjanac V.; Crawley D. B. (1997): The Implementation Of Industry Foundation Classes In Simulation Tools For The Building Industry; Proceedings of Building Simulation '97, Volume 1: S. 203-210

Bazjanac V. (2001): Acquisition of Building Geometry in the Simulation of Energy Performance; Proceedings of Building Simulation 2001, Volume 1: S. 305 – 312

Bazjanac V. (2002): Early Lessons from Deployment of IFC Compatible Software, European Community Product and Process Modeling Conference (ECPPM 2003), 13. September 2002 in Portoroz/Slowenien

Bazjanac V. (2003): Improving Building Energy Performance Simulation with Software Interoperabiltiy, Proceedings of Building Simulation '99, Volume 1: S. 87 – 92

Bazjanac V. (2005): Model Based Cost and Energy Performance Estimation During Schematic Design, 22nd CIB – W78 Conference on Information Technology in Construction, 18. – 21. Juli 2005 in Dresden

Beuke K., Möller B., Cramer H. et al. (2000): Bauinformatik – Fachgebiet des Bauingenieurwesens an den deutschsprachigen Universitäten, Denkschrift des Arbeitskreises Bauinformatik, zu beziehen unter: www.bauinf.tu-cottbus.de (Abgerufen am 23.03.2006)

BLIS Project (2003): SABLE Project Discription; zu beziehen unter: www.blis-project.org/~sable (Abgerufen am 06.03.2006)

Bos B. (1999): XML in 10 Points; World Wide Web Consortium, zu beziehen unter: http://www.w3c.de/Misc/XML-in-10-points.html (Abgerufen am 10.01.2006)

Buhl F. (1999): DOE – 2 Weather Processor, LBNL Simulation Research Group, zu beziehen unter: http://gundog.lbl.gov/ (Abgerufen am 15.01.2006)

Crawley D., Winkelmann F., Lawrie L., Pedersen C. (2001): EnergyPlus: New Capabilities in a Whole – Building Energy Simulation Program, Proceedings of Building Simulation 2001, Volume 1: S. 51 – 58

Dayal M. (2004): Analyse des 3D – Datenaustausches via IFC – Modell am Beispiel komplexer Objektdokumentation in der Automobilindustrie mit dem Ziel der Optimierung von Planungsprozessen; Diplomarbeit an der Technischen Universität München

Depecker P., Menezo C., Virgone J., Lepers S. (2001): Design of Buildings Shape and Energetic Consumption, Building and Environment 36 (2001), S. 627 – 635

FIRST (2006): SMILE – Software für die energetische Gebäude- und Anlagensimulation, Produktblatt; Fraunhofer – Institut für Rechnerarchitektur und Softwaretechnik FIRST

Fischer M., Kam C. (2002): PM4D Final Report, CIFE Technical Report Number 143, Stanford University, USA

Forschungszentrum Karlsruhe (2006): Analysesoftware IfcStoreyView, IfcViewer und IfcObjectcounter; zu beziehen unter: www.iai.fzk.de (Abgerufen am 03.04.2006)

Grabowski H., Anderl R., Polly A. (1993): Integriertes Produktmodell, Entwicklungen zur Normung von CIM; 1. Auflage, Beuth Verlag, Berlin Wien Zürich

Haas W. (1999): Datenaustausch und Datenintegration – STEP und IAI als Beiträge zur Standardisierung, BundesBauBlatt, November 1999

Haller H.-W. (1994): Ein Produktmodell für den Stahlbau, Dissertation an der Fakultät für Bauingenieur- und Vermessungswesen der Universität Karlsruhe

Hitchcock R. J. (2002): Software Interoperability for Energy Simulation, ASHRAE Transactions, Volume 109(1), Jänner 2003

Hörenbaum Ch. (2002): Ein Produktmodell für den Komplettbau, Dissertation an der Fakultät für Bauingenieur- und Vermessungswesen der Universität Karlsruhe

IAI (2006a): IFC Model Development, International Alliance for Interoperability, zu beziehen unter: http://ce.vtt.fi/iaiIFCprojects/ (Abgerufen am 06.03.2006)

IAI (2006b): Online Documentation IFC2x Edition 3, International Alliance for Interoperability; zu beziehen unter: http://www.iai-international.org/Model/R2x3_final/index.htm (Abgerufen am 05.06.2006)

IAI (2006c): IFC kompatible zertifizierte Produkte, International Alliance for Interoperability, zu beziehen unter: http://www.buildingsmart.de/2/2_01_01.htm (Abgerufen am 05.04.2006)

IAI (2006d): Download des im Anwenderhandbuch BIM/IFC verwendeten Beispielprojekts, zu beziehen unter: http://www.buildingsmart.de/anwenderhandbuch.htm (Abgerufen am 17.05.2006)

ISO 10303 (1994): Industrial Automation Systems and Integration – Product Data Representation and Exchange; International Organization for Standardization

ISO 10303-11 (2004): Part 11: Description Methods: The EXPRESS Language Reference Manual; International Organization for Standardization

ISO 16739 (2005): Industry Foundation Classes, Release 2x, Platform Specifications; International Organization for Standardization

Jokela M.; Keinanen A.; Lahtela H.; Lassila K. (1997): Integrated Building Simulation Tool – RIUSKA; Proceedings of Building Simulation '97, Volume 3: X-1-6

Juli R. (2005): Neuer Trend oder alter Hut? Building Information Modeling – Was verbirgt sich wirklich dahinter?, CAD News Magazin, 02/2005; zu beziehen unter: www.cad-news.de (Abgerufen am 21.03.2006)

Karola A., Lahtela H., Hänninen R., Hitchcock R., Chen Q., Dajka S., Hagström K. (2001): BSPro COM Server – Interoperability between Software Tools Using Industry Foundations Classes; Proceedings of Building Simulation 2001, Volume 1: S. 747 - 754

Karola A. (2006): Hintergrundinformationen zu RIUSKA, persönliche Mitteilung

Khemlani L. (2004) The IFC Building Model: A Look Under the Hood, zu beziehen unter: www.aecbytes.com (Abgerufen am 03.04.2006)

Kiviniemi A., Fischer M., Bazjanac V. (2005): Integration of Multiple Product Models: IFC Model Servers as a Potential Solution, 22nd CIB – W78 Conference on Information Technology in Construction, 18. – 21. Juli 2005 in Dresden

Korpowski R. (2003): Stabilität der Objektstruktur des Produktdatenmodells und des Informationsgehaltes des Planungswissens bei IFC2x; Diplomarbeit an der Technischen Universität Dresden, Fakultät Bauingenieurwesen

LBNL Lawrence Berkeley National Laboratory: What Is DOE-2?; zu beziehen unter: http://gundog.lbl.gov/ (Abgerufen am 08.02.2006)

LBNL Lawrence Berkeley National Laboratory: DOE 2.1E Basics Manual; zu beziehen unter: http://gundog.lbl.gov/ (Abgerufen am 15.01.2006)

Liebich T. (Hrsg.), (2004): IFC2x Edition 2 Model Implementation Guide Version 1.7, International Alliance for Interoperability (IAI); zu beziehen unter: www.building smart.de (Abgerufen am 07.02.2006)

Liebich T. (Hrsg.), Hoffeller T. (Hrsg.), (2006): Anwenderhandbuch Datenaustausch BIM/IFC, Industrieallianz für Interoperabilität e. V., zu beziehen unter: www.buildingsmart.de (Abgerufen am 07.04.2006)

Liebich T. (2006): VDI Richtlinien und IFC, persönliche Mitteilung

Nisbet N. (Hrsg.), Liebich T. (Hrsg.), (2005): ifcXML Implementation Guide Version 1.0, International Alliance for Interoperability (IAI); zu beziehen unter: www.buildingsmart.de (Abgerufen am 07.02.2006)

Nour M., Beucke K. (2005): Manipulating IFC Model Data in Conjunction with CAD, 22nd CIB – W78 Conference on Information Technology in Construction, 18. – 21. Juli 2005 in Dresden

Mossack P. (2005): Neuer Trend oder alter Hut? Building Information Modeling – Was verbirgt sich wirklich dahinter?, CAD News Magazin, 02/2005; zu beziehen unter: www.cad-news.de (Abgerufen am 21.03.2006)

Müns M. und Amend B. (2005): Neuer Trend oder alter Hut? Building Information Modeling – Was verbirgt sich wirklich dahinter?, CAD News Magazin, 02/2005; zu beziehen unter: www.cad-news.de (Abgerufen am 21.03.2006)

Olof Granlund Oy (2005): Software Help RIUSKA

Olof Granlund Oy (2006): BSPro COM-Server for IFC-Files; zu beziehen unter: http://193.229.253.197/bspro/bspro_main.htm (Abgerufen am 26.01.2006)

Open Design Alliance, (2006): About the Alliance; zu beziehen unter: www.open design.com (Abgerufen am 23.03.2006)

Plume J., Mitchell J. (2005): A Multi – Disciplinary Design Studio using a Shared IFC Building Model; Veröffentlicht in: Martens B. (Hrsg.), Brown A. (Hrsg.): Computer Aided Architectural Design Futures 2005 – Proceedings of the 11th International CAAD Futures Conference, 20. bis 22. Juni 2005, Technische Universität Wien

Richtlinien CAD Hochbau (1998) des Magistrats der Stadt Wien, Ausgabe 1998

Richtlinie CAD Hochbau (2004) des Österreichischen Bundesministeriums für Landesverteidigung, Baudirektion – Heeresbau- und Vermessungsamt, September 2004

Schinnerl S. (2002): Analyse und Bewertung der Berechnungsalgorithmen der Softwareprogramme DOE-2 und Solar Computer, Diplomarbeit an der Fachhochschule Pinkafeld

Schinnerl S., Zapfel W., Görtler G., Geyer J. (2003): Gebäudesimulationsprogramme im Vergleich, TAB 2/2003, S. 49 – 52

Schwaiger R. (2006): Vereinfachtes ArchiCAD – IFC – Modell, persönliche Mitteilung

Solibri (2006): Accelerating BIM Benefits: Solibri Model Checker, Produktinformation; zu beziehen unter: www.solibri.com (Abgerufen am 21.03.2006)

TRNSYS (2006): SimCAD for TRNSYS: CAD – Tool für Gebäudesimulation mit TRNSYS; zu beziehen unter: http://www.trnsys.de/ts/software/simcad/simcad.htm (Abgerufen am 13.04.2006)

VDI 3805, Blatt 1 (2001): Produktdatenaustausch in der TGA – Grundlagen, Verein Deutscher Ingenieure, Beuth Verlag, Berlin

VDI 6020, Blatt 1 (2001): Anforderungen an Rechenverfahren zur Gebäude- und Anlagensimulation – Gebäudesimulation; Verein Deutscher Ingenieure, Beuth Verlag, Berlin

VDI 6021 (2000): Datenaustausch für die thermische Lastberechnung von Gebäuden; Verein Deutscher Ingenieure, Beuth Verlag, Berlin

VDI 6027, Blatt 1 (1999): Anforderungen an den Datenaustausch von CAD – Systemen, Gebäude- und Gebäudetechnik – Konventionen, Verein Deutscher Ingenieure, Beuth Verlag, Berlin

VDI 6027, Blatt 2 (2005): Anforderungen an den Datenaustausch von CAD – Systemen, Anlagentechnik; Verein Deutscher Ingenieure, Beuth Verlag, Berlin

VDI Richtlinienausschuss 3805 (2006): Produktdatenaustausch in der TGA; zu beziehen unter: www.vdi.de (Abgerufen am: 03.04.2006)

11 Abbildungsverzeichnis

ABBILDUNG 2.1: STRUKTUR DER LOKALEN ZUORDNUNG VON 5
ABBILDUNG 2.2: ALLGEMEINE STRUKTUR DES ANLAGENNETZES (NACH VDI 6027) 5
ABBILDUNG 3.1: FUNKTIONEN DES PRODUKTMODELLS IN DER DATENVERARBEITUNG 10
ABBILDUNG 3.2: GEOMETRISCHES DATENMODELL (NACH KHEMLANI, 2004) 13
ABBILDUNG 3.3: GEBÄUDEDATENMODELL (NACH KHEMLANI, 2004) 14
ABBILDUNG 3.4: DEFINITION EINER EINFACHEN VERERBUNGSSTRUKTUR IN EXPRESS 15
ABBILDUNG 3.5: DARSTELLUNG DES BEISPIELS IN EXPRESS – G 16
ABBILDUNG 4.1: EINSATZMÖGLICHKEITEN DER IFC- SCHNITTSTELLE IN DER ENTWURFSPHASE ... 19
ABBILDUNG 4.2: AUFBAU DER IFC LAYER 20
ABBILDUNG 4.3: IFCROOT SUBTYPE TREE 22
ABBILDUNG 4.4: IFCOBJECT SUBTYPE TREE 23
ABBILDUNG 4.5: RAUMORDNUNG IM IFC – MODELL 24
ABBILDUNG 4.6: AUFBAU EINER STEP DATEI (NACH NOUR, 2005) 24
ABBILDUNG 4.7: BEISPIEL EINER IFC – DATEI (AUSZUG) 25
ABBILDUNG 4.8: IFC VERERBUNGSBAUM 25
ABBILDUNG 4.9: DOKUMENTTYP – DEFINITION IN XML (HÖRENBAUM, 2002) 26
ABBILDUNG 4.10: AUSTAUSCHDATEI ALS XML – DOKUMENT (HÖRENBAUM, 2002) 27
ABBILDUNG 4.11: IFC KOMPATIBILITÄT (HITCHCOCK, 2002) 29
ABBILDUNG 4.12: MÖGLICHKEITEN DES DATENAUSTAUSCHES 30
ABBILDUNG 4.13: GETEILTES DATENVERWALTUNGSSYSTEM 31
ABBILDUNG 4.14: BSPRO COM – SERVER IM DATENFLUSS (KAROLA ET AL., 2001) 32
ABBILDUNG 4.15: SCHEMA DES IFC2X3 MODELLS (NACH IAI, 2006B) 34
ABBILDUNG 4.16: SERVERZUGRIFF OHNE STANDARDISIERTES INTERFACE (VEREINFACHT NACH BLIS PROJECT, 2003) 35
ABBILDUNG 4.17: SERVERZUGRIFF MIT SABLE (VEREINFACHT NACH BLIS PROJECT, 2003) 36
ABBILDUNG 4.18: IAI LOGO 38
ABBILDUNG 5.1: ERSTELLUNG VON TEILFLÄCHEN IM ASPEKTMODELL 45
ABBILDUNG 5.2: STRUKTURMODELL DES THERMISCHEN GEBÄUDEMODELLS (VDI 6021) 46
ABBILDUNG 5.3: KENNZEICHNUNG DES STRUKTURMODELLS NACH VDI 6021 46
ABBILDUNG 5.4: VERTEILUNG DER GESAMTKOSTEN EINER ENERGIESIMULATION (BAZJANAC, 2001) 47
ABBILDUNG 5.5: VERTEILUNG DES AUFWANDS BIS ZUM ERSTEN ERFOLGREICHEN SIMULATIONSDURCHGANG (BAZJANAC, 2001) 48

ABBILDUNG 5.6: ANTEIL DER GEBÄUDEGEOMETRIE AN DER AUFBEREITUNG DER EINGABEDATEN (BAZJANAC, 2001) 48
ABBILDUNG 5.7: DEFINITION DER RAUMBEGRENZUNGEN IM BIM (BAZJANAC, 2005) 53
ABBILDUNG 5.8: SCHNITTSTELLEN UND DATENTRANSFERMÖGLICHKEITEN ZU EINEM BIM (NACH BAZJANAC, 2005) 54
ABBILDUNG 6.1: PROGRAMMABLAUF DOE-2 59
ABBILDUNG 6.2: RESPONSE- UND GEWICHTSFAKTOREN (DEPECKER ET AL., 2001) 61
ABBILDUNG 6.3: DOE – OUTPUT EINER BEHAGLICHKEITSSIMULATION (AUSZUG) 66
ABBILDUNG 6.4: DOE – OUTPUT DER ENERGIESIMULATION (AUSZUG) 66
ABBILDUNG 6.5: SIMULATIONSMÖGLICHKEITEN IN RIUSKA 68
ABBILDUNG 7.1: KELLERGESCHOß (IAI, 2006D) 71
ABBILDUNG 7.2: ERGESCHOß (IAI, 2006D) 71
ABBILDUNG 7.3: OBERGESCHOß (IAI, 2006D) 71
ABBILDUNG 7.4 DACHGESCHOß (IAI, 2006D) 71
ABBILDUNG 7.5: IFC – MODELL VON ADT 72
ABBILDUNG 7.6: IFC – MODELL VON ARCHICAD 72
ABBILDUNG 7.7 IFC – MODELL VON ALLPLAN 73
ABBILDUNG 7.8: IFC – MODELL VON BENTLEY 73
ABBILDUNG 7.9: IFC – MODELL VON REVIT 73
ABBILDUNG 7.10: ADT – MODELL IN DDS 75
ABBILDUNG 7.11: EG – GRUNDRISS (ADT) 75
ABBILDUNG 7.12: ADT – MODELLS IN DDS (2) 75
ABBILDUNG 7.13: ARCHICAD – MODELL IN DDS 76
ABBILDUNG 7.14: EG – GRUNDRISS (ARCHICAD) 76
ABBILDUNG 7.15: ALLPLAN – MODELL IN DDS 76
ABBILDUNG 7.16: EG – GRUNDRISS (ALLPLAN) 76
ABBILDUNG 7.17: BENTLEY – MODELL IN DDS 77
ABBILDUNG 7.18: EG – GRUNDRISS (BENTLEY) 77
ABBILDUNG 7.19: REVIT – MODELL IN DDS 77
ABBILDUNG 7.20: EG – GRUNDRISS (REVIT) 77
ABBILDUNG 7.21: REVIT – MODELL IN DDS (2) 77
ABBILDUNG 7.22: ANSICHT DES DDS – MODELLS IN IFCSTOREYVIEW 78
ABBILDUNG 7.23: ADT – MODELL IN RIUSKA 4.0.13 80
ABBILDUNG 7.24: ARCHICAD – MODELL IN RIUSKA 4.0.13 81
ABBILDUNG 7.25: ALLPLAN – MODELL IN RIUSKA 4.0.13 81

ABBILDUNG 7.26: BENTLEY – MODELL IN RIUSKA 4.0.13 81
ABBILDUNG 7.27: REVIT – MODELL IN RIUSKA 4.0.13 82
ABBILDUNG 7.28: DDS – MODELL 82
ABBILDUNG 7.29: GRUNDRISS EG (ALLPLAN) 83
ABBILDUNG 7.30: GRUNDRISS EG (BENTLEY) 83
ABBILDUNG 7.31: ADT – MODELL IN RIUSKA 4.0.26 84
ABBILDUNG 7.32: ARCHICAD – MODELL IN RIUSKA 4.0.26 84
ABBILDUNG 7.33: ALLPLAN – MODELL IN RIUSKA 4.0.26 84
ABBILDUNG 7.34: BENTLEY – MODELL IN RIUSKA 4.0.26 84
ABBILDUNG 7.35: REVIT – MODELL IN RIUSKA 4.0.26 84
ABBILDUNG 7.36: DDS – MODELL IN RIUSKA 4.0.26 84
ABBILDUNG 7.37: VEREINFACHTES ARCHICAD – MODELL IN RIUSKA 86
ABBILDUNG 8.1:GEBÄUDEDATENMODELL (NACH KHEMLANI, 2004) 92
ABBILDUNG 8.2: STRUKTUR DER IFC – LAYER 93
ABBILDUNG 8.3: VERTEILUNG DER GESAMTKOSTEN EINER ENERGIESIMULATION (BAZJANAC, 2001) 94
ABBILDUNG 8.4: ANTEIL DER GEBÄUDEGEOMETRIE AN DER AUFBEREITUNG DER EINGABEDATEN (BAZJANAC, 2001) 94
ABBILDUNG 8.5: DDS – MODELL IN RIUSKA (VERSION 4.0.26) 96
ABBILDUNG 8.6: REVIT – MODELL IN RIUSKA (VERSION 4.0.13) 96
ABBILDUNG 8.7: REVIT – MODELL IN RIUSKA (VERSION 4.0.26) 96

12 Tabellenverzeichnis

TABELLE 2.1: VDI 3805: PRODUKTDATENAUSTAUSCH IN DER TGA – FOLGEBLÄTTER 7
TABELLE 6.1: VERGLEICH ALLGEMEINER FEATURES 63
TABELLE 6.2: VERGLEICH VON ANLAGENSPEZIFISCHEN FEATURES 63
TABELLE 6.3: VERGLEICH VON LASTSPEZIFISCHEN FEATURES 64
TABELLE 7.1: ERGEBNISSE DES IMPORTS DER IAI – MODELLE IN DDS 78
TABELLE 7.2: ERGEBNISSE DES IMPORTS DER IFC – MODELLE IN RIUSKA 4.0.13 83
TABELLE 7.3: ERGEBNISSE DES IMPORTS DER IFC – MODELLE IN RIUSKA 4.0.26 85
TABELLE 7.4: ERGEBNISSE DER INFORMATIONSÜBERTRAGUNG IN DIE IFC – MODELLE 87
TABELLE 7.5: PSET EINES RAUMS IM DDS-RAUMDATENDIALOG 88

Eidesstattliche Erklärung

Ich erkläre hiermit, dass ich die vorliegende Diplomarbeit selbständig und nur unter Verwendung der angegeben Literatur und Hilfsmittel erstellt habe.

Pinkafeld, im Juni 2006

Carina Sagerschnig